PGS. TS. NGUYỄN VIẾT TRUNG (Chủ biên) TS. HOÀNG HÀ - KS. LÊ QUANG HANH

TÍNH TOÁN Kỹ thuật xây dựng trên excel

(Tái bản)

NHÀ XUẤT BẢN XÂY DỰNG HÀ NỘI - 2012

LỜI NÓI ĐẦU

Trong nhiều năm giảng dạy môn "Tin học ứng dụng" cho sinh viên và học viên Cao học ở Đại học Giao thông Vận tải, tác giả luôn cảm thấy bị một sức ép là càng ngày càng có quá nhiều kiến thức "Tin học ứng dụng" phải được chuyển tải đến các kỹ sư và sinh viên để họ kịp bắt nhịp được với đòi hỏi thực tế sản xuất và thị trường. Vậy mà thời gian để dạy và học chỉ có hạn vì còn nhiều môn học khác. Có lẽ đó là vì ngành Tin học là một trong vài ngành mà kiến thức đổi mới quá nhanh sau mỗi một năm.

Một câu hỏi thường được nêu ra là: Liệu có hợp lý hay không khi yêu cầu người sinh viên hay kỹ sư phải học để biết cách sử dụng các bộ chương trình thông thường về tính toán kết cấu, cơ cấu máy, chi tiết máy, lập dự toán, lập tiến độ thi công, quản lý dự án, vẽ kỹ thuật, v,v... do người khác viết ra, rồi lại phải học một hay vài ngôn ngữ lập trình như PASCAL, C, C++, BASIC để tự mình viết các ứng dụng nhỏ riêng cho công tác hàng ngày. Trong khi đó thì các bộ phần mềm thảo chương như TURBO PASCAL, DELPHI, VISUAL BASIC, VISUAL C, v.v... cứ thay đổi phát triển đến chóng mặt.

Ngày nay hầu như mọi kỹ sư và cán bộ đều phải biết dùng ở mức độ nào đó bộ phần mềm MICROSOFT OFFICE trong công tác hàng ngày để viết báo cáo, quản lý dữ liệu, soạn thảo văn bản, thư từ, gửi fax, tính toán chi tiêu, v.v... Trong bộ OFFICE, phần mềm EXCEL giữ một ví trí quan trọng nhưng hầu như còn ít được sử dụng cho các tính toán kỹ thuật và khoa học mà chủ yếu dùng cho công tác tài vụ, kinh tế, quản trị.

Khi làm việc với các Chuyên gia Tư vấn của Ngân hàng thế giới (WB) và của Ngân hàng phát triển cháu Á (ADB), tác giả đã được họ khuyên nên tự mình sử dụng và hướng dẫn cho các sinh viên dùng EXCEL để giải phần lớn các bài toán thông thường trong thiết kế kết cấu cầu - đường và giải các bài toán khoa học kỹ thuật cho nhiều ngành khác. Trên thực tế để phục vụ lập Dự án 6 cầu đường sắt trên tuyến Hà Nội - Hồ chí Minh, Dự án 38 cầu trên Quốc lộ 1, và nhiều Dự án cầu - đường khác, các kỹ sư Việt Nam và nước ngoài đã tính toán nhiều vấn để trên EXCEL.

Cuốn "Tính toán kỹ thuật xây dựng trên Excel" được viết ra với những suy nghĩ như trên, nhằm phục vụ các sinh viên và kỹ sư giải các bài toán khoa học và kỹ thuật thường gặp trên EXCEL để trong đa số các tình thuống có thể thay cho việc họ phải tốn công sức và thời gian học cho nắm vững và viết được các chương trình bằng một ngôn ngữ lập trình như PASCAL hay C. Mặt khác kết quả tính toán bằng EXCEL được trình bày đẹp mắt và dễ dàng trình duyệt lên cấp trên. Tất nhiên nếu nắm vững các ngôn ngữ như PASCAL hoặc C thì bạn đọc sẽ có khả năng làm việc tốt hơn nữa.

Sách được biên soạn lần đầu tiên nên không tránh khổi các thiếu sót. Tác giả xin chân thành cám ơn và sẵn sàng tiếp thu mọi ý kiến phê bình của bạn đọc để hoàn thiện thêm cuốn sách này. Nhiều chương trình mẫu trong sách này đẳ có sẵn trên đĩa. Bạn đọc nào cần chúng hãy liên hệ với Nhà xuất bản Xáy dựng và tác giả để sao đĩa cho nhanh.

Các tác giả

Chương 1

CÁC PHÉP TOÁN KHOA HỌC KỸ THUẬT THỰC HIỆN TRÊN EXCEL

Ngày nay Excel đã trở nên quá quen thuộc để giải các bài toán kinh doanh, tài chính, kế toán thường nảy sinh trong các cơ quan và doanh nghiệp, vì vậy các bạn đọc là kỹ sư hay nhà khoa học sẽ có thể đặt ra các câu hỏi nghi ngờ như:

- Độ chính xác của các phép toán trên Excel có thoả mãn yêu cầu của các tính toán khoa học hay không?

- Các hàm có sẵn trong Excel có đủ và phù hợp với các yêu cầu của việc tính toán phục vụ cho khoa học hay không?

- Có thể xây dựng các thuật toán hữu ích, thiết thực trên Excel hay không?

Có thể khẳng định trả lời ngay là Excel đủ công cụ tính toán và độ chính xác tính toán để đáp ứng các tính toán khoa học kỹ thuật thông dụng trong mọi ngành kỹ thuật. Ngoài ra, số lượng các hàm có sẵn của Excel còn nhiều hơn một số ngôn ngữ lập trình thông dụng khác như Pascal chẳng hạn.

1.1. ĐỘ CHÍNH XÁC CỦA CÁC PHÉP TOÁN TRÊN FXCEI.

Độ chính xác của các phép toán trên Excel và độ lớn của trị số xử lý được trong Excel hoàn toàn thoả mãn các tính toán kỹ thuật. Chúng ta đều còn nhớ rằng ngay trước khi máy tính điện tử ra đời thì thước tính cầm tay đã từng là công cụ chính của mọi kỹ sư trên thế giới và độ chính xác của thước tính lấy đến 3 số lẻ sau dấu thập phân tất nhiên không thể so sánh với máy vi tính được.

Tuy nhiên, nhiều thuật toán của phương pháp số rất nhạy cảm với độ chính xác làm toán số học của công cụ tính toán, đặc biệt là khi tính toán theo sơ đồ sai phân hoặc giải bằng số các phương trình vi phân.

Các phép tính thiên văn và khí tượng thường đòi hỏi độ chính xác rất cao.

1.1.1. Độ chính xác số học

Excel cung cấp độ chính xác số học đáp ứng hoặc vượt quá so với khả năng của một số công cụ tính toán thông dụng trong khoa học kỹ thuật. Excel lưu trữ trị số với 15 chữ số. Các máy tính tay dùng trong khoa học thường chỉ thể hiện được 10 chữ số. Máy tính mini điển hình là máy VAX chỉ lưu giữ 7 chữ số với độ chính xác đơn, dấu phẩy động

và lưu giữ 15 chữ số với độ chính xác kép. Siêu máy tính điển hình là Cray-1 chỉ chứa 15 chữ số với độ chính xác đơn, dấu phẩy động.

Excel lưu giữ 15 chữ số trong bộ nhớ nhưng làm tròn số rồi thể hiện lên màn hình trị số tuỷ theo định dạng của ô đã được chính người sử dụng quy định trong bảng tính. Nếu muốn lưu giữ số trong bộ nhớ đúng như con số hiện ra trong ô bảng tính thì hãy chọn lệnh *Calculation* từ thực đơn *Option* rồi chọn hộp kiểm tra với mục *Precision as Displayed* trong hộp hội thoại *Calculation Options*. Nhiều khi nên giảm bớt độ chính xác của phép tính vì như vậy tốc độ tính toán sẽ tăng lên nhiều. Điều này cũng nên làm khi bạn tính tiền mà chỉ xét đến đơn vị đồng chứ không muốn chú ý đến đơn vị hào hay xu.

Xin nhắc lại về cách định dạng chữ số hiện ra trong ô. Hãy dùng chuột để chọn theo trình tự sau:

Format => Cells... => Number (Category) => Code: rồi nhập vào hộp Code những chữ số 0 bằng tổng số chữ số mà bạn muốn hiện ra trong ô, nhưng như trên đã nói: chúng ta chỉ thể hiện được nhiều nhất là 15 chữ số thôi. Nếu cứ cố nhập nhiều hon nữa thì từ chữ số thứ 16 trở đi, excel sẽ làm tròn số và coi như số 0.

| Ví dụ nếu chúng ta nhập số | 12345678901234567890 |
|----------------------------|--------------------------|
| thì máy hiểu là | 123456789012346000000./. |

Điều này có thể làm giảm mất độ chính xác của các phép toán với các số có quá nhiều chữ số.

Với cách làm tròn như vậy mà trong quá trình tính toán thì máy sẽ thực hiện vô số phép tính lập cho nên có thể dự báo rằng kết quả cuối cùng có thể sai khá nhiều.

1.1.2. Giới hạn các trị số trong Excel

- Excel xử lý các trị số trong khoảng từ $2,226 \times 10^{-308}$ đến 1798×10^{-308} .

- Một máy tính bấm tay thường chứa số cỡ 10^{±99}.

- Máy tính mini VAX chứa các số cỡ $10^{\pm 38}$ với độ chính xác đơn và cỡ $10^{\pm 308}$ với độ chính xác kép.

- Siêu máy tính Cray-1 chứa các số cỡ $10^{\pm 2500}$.

Bạn đọc có thể tự rút ra nhận xét khi so sánh các con số nói trên.

Mặc dù Excel lưu giữ được con số lớn nhất là $1798 \times 10^{\pm 308}$ nhưng con số lớn nhất mà chúng ta có thể đánh vào máy tính chỉ là $9,999 \times 10^{\pm 307}$. Nếu ta cố tình nhập vào số lớn hơn thì Excel sẽ hiểu đó là chuỗi ký tự chứ không phải là con số.

Thực ra hầu hết các tính toán khoa học kỹ thuật chỉ liên quan đến các con số trong khoảng 10⁻⁴⁰ đến 10⁺⁴⁰. Tuy nhiên khi các con số này được dùng đến trong phương trình thì các kết quả tính trung gian có thể quá lớn đến mức vượt quá khả năng lưu giữ của

máy tính và kết quả sẽ sai. Ví dụ xét biểu thức đơn giản của cơ học lượng tử là: $\frac{2m}{b^2}$; trong đó:

h - hằng số Planck chia cho 2π (1,0546 × 10⁻³⁴ J-s).

m - khối lượng điện tử còn lại $(9,11 \times 10^{-31} \text{ kg})$.

Kết quả của phép tính này là $1,64 \times 10^{-38}$, đó vẫn còn là con số kích cỡ hợp lý. Nhưng kết quả trung gian của phép tính bình phương rồi nghịch đảo đã là $8,99 \times 10^{-67}$ nghĩa là lớn gấp rất nhiều lần so với kết quả cuối cùng. Máy tính bấm tay cũng sẽ cho kết quả sai trong bài toán đơn giản này.

Hiên tượng tràn ô nhớ dẫn đến kết quả tính toán sai thực ra là bắt đầu từ sai lầm của người đặt ra bài toán mà không hiểu rõ bài toán của chính mình. Cách tránh tình trang này rất đơn giản là phân chia nhỏ và xắp xếp quá trình tính toán sao cho không bao giờ nàv sinh kết quả tính trung gian quá lớn đến nỗi tràn ô nhớ.

Với đô lưu trữ số đến $10^{\pm 308}$ của Excel thì thực tế chúng ta sẽ không gặp phải vấn đề tràn ô nhớ. Tuy nhiên, nếu ban cố tình tao ra một số thật lớn thì Excel sẽ đánh dấu ô đó với ký hiệu #NUM! để thông báo cho ban đừng dùng số liệu đó nữa. Ngược lại, nếu ban thử tao ra một số rất bẻ tức là nhỏ hơn $2,226 \times 10^{-308}$ thì Excel sẽ lưu trữ nó như số 0.

1.1.3. Lõi trong Excel

Excel sẽ thông báo về 7 trường hợp lỗi như sau:

- 1. # DIV/0! chia cho số 0.
- 2. # NAME? chưa định nghĩa tên biến trong ô.
- 3. # N/A không có trị số nào sẵn có cho tình huống đang xét.
- 4. # NULL! kết quả chẳng có gì cả.
- 5. # NUM! tràn ô nhớ hoặc dùng tham số vô nghĩa, ví du SQRT(-1).
- 6. # REF! tham chiếu ô không có giá tri, ô này không có trên bảng tính.
- 7. # VALUE! kiểu của đối số không đúng, ví dụ lẽ ra là con số thì trong ô lại là ký tự.

Khi gặp một trong các lỗi này thì kết quả sẽ là sai. Sai lầm này sẽ lan truyền đi khắp bảng tính. Vì vậy Excel dùng các thông báo như trên để cảnh báo cho người tính toán.

1.2. KHÁI NIÊM VỀ THAM CHIẾU Ô TRONG BẢNG TÍNH EXCEL

Các ô trong bảng tính Excel có thể chứa con số hoặc ký tư hoặc công thức. Excel đủ thông minh để theo dõi những gì ban đang gõ từ bàn phím và gán cho nó kiểu đúng như ý của bạn. Nếu nội dung ô là trị số thì Excel sẽ lưu giữ nó dưới dạng con số và sau này ban có thể dùng số đó để tính toán. Nếu nội dung ô lẫn lộn cả ký tự và chữ số thì Excel coi đó như là văn bản. Nếu nội dung ô bắt đầu với dấu bằng (=) thì Excel cất giữ theo dạng công thức.

Mọi ô trong bảng tính Excel đều có 2 phần liên kết với nhau chặt chẽ: phần nội dung và phần giá trị. Nội dung là phần mà bạn gõ nhập vào ô, còn giá trị là cái mà bạn nhìn thấy trên màn hình. Việc định dạng ô không ảnh hưởng đến giá trị của ô mặc dù có thể làm thay đổi trị số xuất hiện trên màn hình. Đối với văn bản hay con số thì nội dung và giá trị là giống nhau. Đối với các công thức thì nội dung là công thức mà bạn gõ vào còn giá trị là kết quả tính toán theo công thức đó.

Có thể chèn giá trị của một ô nào đó trong bảng tính vào công thức đang được bạn gõ bằng cách dùng tham chiếu ô. Một tham chiếu ô bao gồm nhóm ký tự và chữ số mà chữ cái chỉ ra tên cột chứa ô tham chiếu còn chữ số chỉ ra tên của dòng tham chiếu. Ví dụ ô B8 sẽ liên quan đến giá trị của ô thuộc cột B và dòng thứ 8 của bảng tính. Bạn cũng có thể tham chiếu theo tên kép cả cột và dòng như R5C7, kiểu tham chiếu này gọi là kiểu R1C1.

Bạn sẽ đặt kiểu tham chiếu ô mà bạn muốn bằng cách đánh dấu kiểm tra hay dấu không kiểm tra vào ô kiểm tra R1C1 trong hộp đối thoại *Workspace*. Để hiện lên hộp đối thoại này, hãy chọn lệnh *Workspace* trên thực đơn *Option*. Nếu bạn tạo ra một bảng tính dùng một kiểu rồi thay đổi sang kiểu R1C1 nhờ hộp kiểm tra R1C1 thì mọi tham chiếu của bạn sẽ đổi sang kiểu khác nói trên.

1.2.1. Tham chiếu ô ngoài

Nếu bạn tham chiếu đến một ô thuộc một bảng tính khác với bảng tính mà bạn đang gõ công thức vào thì bạn phải thêm tên của bảng tính đó vào để Excel biết chỗ mà tham chiếu. Điều này gọi là tham chiếu ngoài. Bảng tính chứa ô tham chiếu ngoài không cần phải được mở. Để tạo tham chiếu ngoài hãy gõ tên bảng tính, rồi gõ dấu chấm than (!), rồi gõ tên ô tham chiếu. Nếu bảng tính không thuộc thư mục mà bạn đang làm việc thì cần phải báo thêm đường dẫn đến thư mục chứa bảng tính tham chiếu. Ví dụ bạn muốn tham chiếu ô S5 trên bảng tính SOLIEU.XLS thuộc thư mục hiện hành, thì bạn sẽ viết như sau:

SOLIEU.XLS ! S5

Nếu bảng tính nói trên lại thuộc thư mục E:\KETCAU chẳng hạn thì bạn sẽ phải tham chiếu đến:

E:\KETCAU\ SOLIEU.XLS !S5

Dấu nháy đơn là phải đưa vào khi có thông báo đường dẫn đến thư mục.

Con đường đơn giản nhất để bảo đảm đúng địa chỉ ô mà bạn muốn tham chiếu trong một bảng tính khác là hãy mở bảng tính đó ra. Sau đó chuyển về bảng tính đang làm rồi gõ công thức vào ô mà bạn muốn thực hiện, khi gõ đến chỗ cần thêm tên ô tham chiếu

thì hãy lại chuyển trở về bảng tính cần tham chiếu rồi nhắp chuột vào ô nào mà bạn muốn tham chiếu. Thế là tên ô đó sẽ tự đông được ghi đúng chỗ bạn muốn trong bảng tính hiện hàch

1.2.2. Tham chiếu vùng ô

Đôi khi chúng ta cần tham chiếu cả một vùng ô. Quy ước một vùng ô là một hình chữ nhật trên bảng tính. Mọi ô nằm trong vùng đó coi như thuộc nội dung của vùng đó (nghĩa là không có ô trống nào). Tên một vùng sẽ gồm tên ô trên cùng bên trái và tên ô dưới cùng bên phải của vùng đó, hai tên này cách nhau bởi dấu hai chấm (:), ví dụ A3:F7.

Bạn có thể tổ hợp nhiều vùng vào trong một tham chiếu ô duy nhất bằng cách thêm dấu phẩy (,) vào giữa tên các vùng. Ví dụ tham chiếu H4:J6, L2:M3 chứa hai vùng là vùng H4:J6 và vùng L2:M3. Khi đó mọi ô trong hai vùng đó đều được tham chiếu (nghĩa là các ô H4, H5, H6, I4, I5, I6, J4, J5, J6, L2, L3, M2 và M3). Nếu bạn gõ dấu cách trống để thay cho dấu phẩy nhằm phân cách hai tên vùng thì kết quả tham chiếu sẽ là những ô nào nằm trong miền giao của hai vùng đó. Ví dụ H4:J6, L2:M3 sẽ chỉ là tham chiếu đến các ô J5 và J6 vì chúng đồng thời nằm trong cả 2 vùng nói trên.

Cũng giống khi tham chiếu ô đơn lẻ, cách thức đơn giản nhất để tham chiếu vùng ô là chọn nó bằng cách nhấn và rê chuột trong khi đang gõ công thức vào ô.

1.2.3. Tham chiếu ô tương đối và tham chiếu ô tuyệt đổi

1.2.3.1. Tham chiếu ô tương đối

Phần lớn các tham chiếu ô mà chúng ta thường dùng đều là tham chiếu tương đối. Nó nói lên mối tương quan vị trí giữa ô đang xét và ô được tham chiếu. Ví dụ công thức trong ô G5 có chứa tham chiếu ô E3. Khi đó ký hiệu E3 không thực sự cứng nhắc là tham chiếu đến nội dung của ô thuộc cột E và dòng thứ 3 mà là đại diện cho nội dung ô nằm cách ô G5 hai cột về bên trái và nằm cách ô G5 hai dòng về phía trên. Chẳng hạn, nếu bạn sao chép công thức của ô G5 vào ô I8 thì bạn sẽ thấy ngay là ô tham chiếu bây giờ sẽ là ô G6, ô này nằm cách ô I8 hai cột về bên trái và cách hai dòng về phía trên.

Trong kiểu tham chiếu R1C1, bạn sẽ đặt các dấu ngoặc vuông bao ra ngoài số hiệu cột và dòng, điều này sẽ làm cho chúng thay đổi cả về hướng và về khoảng cách. Điểm gốc của hệ thống tham chiếu ô là góc trái trên của bảng tính, các hướng dương là hướng xuống dưới và hướng sang phải. Ví dụ R[-2]C[2] là tham chiếu ô tương đối đến ô nằm cách 2 dòng về phía trên và nằm cách 2 cột về phía bên phải của ô đang xét.

Kiểu tham chiếu ô tương đối rất tiện lợi khi bạn muốn dùng 1 công thức để xử lý cả dãy số liệu cùng kiểu. Giả sử bạn đang tính toán theo cùng 1 công thức với 50 bộ số liệu ban đầu khác nhau, đáng lẽ phải gõ 50 lần công thức đó thì bây giờ bạn chỉ gõ 1 lần công thức vào ô của dòng đầu tiên. Sau đó nhờ việc sao chép sang các ô khác trong cùng

cột đó, Excel sẽ thực hiện việc tham chiếu ô tương đối và bạn sẽ hoàn thành công việc sau vài lần nhắp chuột.

1.2.3.2. Tham chiếu ô tuyệt đối

Một tham chiếu ô tuyệt đối sẽ không thay đổi khi bạn sao chép công thức, nó luôn tham chiếu nội dung của ô đã được chỉ định. Bất kể bạn sao chép công thức đến ô nào trong bảng tính hiện hành. Để tham chiếu ô tuyệt đối bạn chỉ cần thêm dấu \$ trước tên cột hay tên dòng của ô tham chiếu. Ví dụ \$G\$5 là tham chiếu tuyệt đối đến ô G5.

Kiểu tham chiếu R1C1 luôn là tham chiếu tuyệt đối, trừ khi bạn dùng các dấu ngoặc để đánh dấu đó là tham chiếu tương đối. Ví dụ R5C7 là tham chiếu đến các ô nằm ở vùng giao của dòng 5 và cột 7.

Tham chiếu tuyệt đối rất hữu ích khi bạn muốn tham chiếu các hệ số trong các công thức mà bạn đang sao chép. Các hệ số này lại có giá trị đã được quy định không thay đổi từ trước. Như vậy khi lập bảng tính cho một bài toán kỹ thuật, bạn hãy đặt các hệ số vào một số ô rồi sau này khi viết công thức và sao chép công thức sẽ tham chiếu tuyệt đối đến các ô đó. Bằng cách này bạn còn có thể theo dõi các diễn biến kết quả khi vẫn dùng cùng công thức nào đó nhưng cho tính toán với các giá trị khác nhau của các hệ số.

1.2.3.3. Tham chiếu ô hỗn hợp

Một tham chiếu hỗn hợp sẽ vừa tham chiếu tương đối đến tên cột hoặc tên dòng và vừa tham chiếu tuyệt đối đến tên cột hay tên dòng còn lại.

Ví dụ, tham chiếu \$G5 sẽ cố định xét cột G nhưng số hiệu dòng thì có thể thay đổi trong lúc sao chép công thức. Trong kiểu tham chiếu R1C1 thì R2C[4] là tham chiếu tuyệt đối đến dòng 2 nhưng tham chiếu tương đối đến cột nằm cách ô đang xét 4 ô về phía bên phải.

1.2.4. Đặt tên cho vùng ô, cho ô riêng lẻ

Có thể đặt tên cho vùng ô hoặc cho 1 ô riêng lẻ rồi dùng tên đó để thay cho việc tham chiếu ô trong các công thức hay các lệnh. Hãy dùng lệnh *Define Name* hay lệnh *Create Names* trong thực đơn *Formula* để tạo ra hoặc thay đổi hoặc xoá hoặc liệt kê các tham chiếu đã được đặt tên. Khi Excel tính toán các công thức mà có chứa các tên thì nó sẽ thay thế mỗi tên bằng định nghĩa tương ứng của tên đó rồi tính toán ra kết quả. Các vùng được đặt tên là các tham chiếu tuyệt đối vì chúng sẽ không thay đổi khi bạn sao chép các ô của vùng.

Sử dụng tên là biện pháp hiệu quả khi bạn phải tham chiếu một ô hay một vùng ô nào đó nhiều lần. Điều quan trọng là các tên làm cho công thức trở nên dễ hiểu hơn vì bạn có thể nhìn vào công thức ghi trong ô nào đó mà suy ra bản chất kỹ thuật của chúng. Ví dụ nếu ô C5 chứa trị số trọng lượng riêng của vật liệu và bạn đã đặt tên là Gamma thì trong

các công thức có tham chiếu đến ô này sẽ chứa từ ngữ Gamma, như vậy dễ hiểu hơn là ghi C5 trong công thức tham chiếu.

Khi bạn gõ tên vùng vào công thức mà công thức này yêu cầu một trị số đơn thì Excel chỉ dùng ô đơn trong vùng đó. Ô mà nó dùng sẽ phụ thuộc vào vị trí tương đối của ô chứa tham chiếu so với vùng mà nó tham chiếu. Nếu tham chiếu đến một cột của các ô ở bên phải hay bên trái của ô chứa tham chiếu thì ô được dùng là ở trên cùng hàng với ô có trong tham chiếu. Tương tự, nếu tham chiếu là dòng trên hay dòng dưới của ô chứa tham chiếu thì ô được dùng là ô nằm trong cùng cột với ô chứa tham chiếu. Nếu bạn có hơn một dòng hay một cột trong tham chiếu trong cùng hàng hay cùng cột như ô tham chiếu thì bạn sẽ thấy hiện trong ô là #VALUE! error. Bạn cũng sẽ thấy tương tự nếu không ô nào trong tham chiếu là ở trong cùng cột hay cùng hàng với ô tham chiếu.

Thực ra, có thể dùng lệnh *Define Name* để đặt tên cho bất kỳ giá trị nào bao gồm cả các con số và các chữ cái mà chúng không chứa trong bất kỳ ô nào. Việc xử lý danh sách các vùng và các trị số đã được đặt tên như là việc thay thế đơn giản một bảng, ở đó giá trị của tên được chèn vào công thức trước khi tính toán theo công thức đó.

Nên gán tên cho ô và vùng ô rồi sử dụng các tên đó trong các công thức. Sử dụng tên thay cho địa chỉ ô tổ ra hiệu quả hơn, vì bạn sẽ tránh phải gõ vào các địa chỉ ô phức tạp. Việc sử dụng tên thay cho các nhãn dòng hay cột trong các công thức ngôn ngữ tự nhiên còn cho phép bạn đặt tên các ô từ bất cứ đâu trong Workbook, hay thậm chí trong các Workbook khác.

Sau khi bạn đã định nghĩa tên trong Worksheet, thì các tên đó có thể được sử dụng trong các Worksheet khác nhau của Workbook.

Tên phải bắt đầu bởi chữ cái hoặc dấu $_(nối dưới), \setminus (sổ chéo ngược), có độ dài nhiều nhất là 255 ký tự và không được chứa dấu cách. Để dễ dọc, các chữ cái đầu mỗi từ trong tên nen viết hoa. Không nên gõ dấu tiếng Việt trong tên.$

Bạn có thể sử dụng tên ô hay tên vùng trong công thức. Kết quả sẽ tương tự như khi sử dụng địa chỉ ô hay địa chỉ vùng.

Vi dụ: Có thể gõ công thức vào ô E3 như sau:

```
=IF(AND(E2>=24,E2<=33),E2-0.6,IF(AND(E2>=15,E2<24),E2-0.5))
```

Néu bây giờ định nghĩa ô E2 có tên là L thì công thức trong ô E3:

=IF(AND(L>=24,L<=33),L-0.6,IF(AND(L>=15,L<24),L-0.5))

sẽ trả về đúng kết quả như trên.

Sau khi đã chọn ô hay vùng ô có hai cách sau:

- Chon Insert \ Name \ Define (hoặc Ctrl + F3).

Trong khung Names in Workbook gõ vào tên cho ô hay nhóm \rightarrow OK.

| Microsoft Excel - Thep1 | | | |
|------------------------------------|----------------------|-----------|------------------------------------|
| Elle Edit View Insert Format Lools | ata <u>Window</u> He | lp | |
| | 1 0.000 | 🔮 Σ f# | 2 2 2 . |
| .VnTime • 11.5 • 18 / | | 国 \$ % | · *:8 +:8 律律 图· &· <u>A</u> ·. |
| Ltt = | =IF(AND(L> | =24,L<=33 |),L-0.6,IF(AND(L>=15,L<24),L-0.5)) |
| Name Box B C | D | E | F G H I |
| 1 I .Số LIÊU TÍNH TOÁN | | | |
| 2 * Chiều dài kết cấu nhịp | L= | 30 | (m) |
| 🛐 🐐 Khẩu độ tính toán | Ltt= | 29.4 | (m) |
| 4 * Khổ Câu | B= | 7 💌 | (m) |
| 5 * Lê người đi một bên | | 1.5 💌 | (m) |
| 6 * Tåi trọng thiết kế: | | H30-XB80 | Tải Trọng Người 300 (kg/m2) |
| 7 * Dâm chủ (dâm định hình tiêu | chuẩn) | 1008 👱 | |
| 8 Số dầm chủ | | 7 | |
| 9 Loại thép làm dầm chủ | Thep Hop Kim Tha | ар 💽 | |
| 10 Orờng độ tính toán của thép | | | |
| 11 Khi chịu lực dọc trục | Ro= | 2700 | (kg/cm2) |
| 12 Khi chịu uốn | Ru= | 2800 | (kg/cm2) |

Hình 1.1: Đặt tên cho cho ớ hay nhóm.

| and the second | A |
|--|------------------|
| | Republic Control |
| 0 | К |
| | |
| Clo | se |
| | |
| <u>A</u> | bt |
| | |
| Del | ete |
| State of the second | M. Ster |
| a post of the second second | 31.5 |
| • | |
| | |
| | 3 |
| | |

Hình 1.2: 1 Cách thứ 1; 2. Cách thứ 2:

- Nháy chuột vào hộp Name Box trên dải công thức (hình 1.1).

Gõ tên vào rồi ấn Enter (không được nháy chuột).

a) Đặt tên theo tiêu đề của cột hay hàng (tự động)

Các bước như sau:

- Chọn ô hay nhóm ô cần đặt tên gồm cả tiêu đề cột hoặc hàng.

- Chon Insert \ Name \ Create (hoặc Ctrl + Shift + F3).

Ý nghĩa các mục trong hộp Create Name sau:

- *Top row:* Lấy ô ở hàng đầu (của khối đã chọn) làm tên.

- Left column: Lấy ô ở cột bên trái (khối đã chọn) làm tên.

- Bottom row: Lấy ô ở hàng cuối (của khối đã chọn) làm tên.

- *Right column:* Lấy ô ở cột bên trái (khối đã chọn) làm tên.

- OK.

b) Dán tên vào công thức

- Khi nhập tên hay sửa công thức, thay vì điển vào địa chỉ của ô (hay miền) đã được đặt tên ta nhấn phím F3 (hoặc nháy chuột vào hộp *Name Box* trên dải công thức hay chọn *Insert* \ *Name* \ *Paste*).

- Chọn tên cần thiết từ danh sách rồi OK.

c) Về nhanh một ô hay miền đã được đặt tên

Có thể đưa con trỏ về nhanh một ô (hay miền) bằng một trong hai cách sau:

Cách thứ I:

- Ân phím F5.

- Chọn tên cần thiết từ danh sách.

- OK.

| 🔀 Microsoft Exc | el - Thep1 | | 计子指生 | · · · · · · · · · · · · · · · · · · · | |
|-----------------|--|-------------------|-------------|---------------------------------------|------|
| Eile Edit Vie | w Insert F <u>o</u> rmat <u>T</u> ool | s Data Window H | lelp | | |
| | 1 a 🦈 🗶 🖻 🖻 | Son. | 🐞 Σ fa | | 100% |
| .VnTime | • 12 • B | 「山屋屋屋 | \$ \$ | % , [*] .0 .00 | 使使 |
| ES | ♥ 분4130 | = =Hs/10 | | | |
| Hs | <u> </u> | D | E | F | G |
| Hsk | NÐINF | LDAMCHU | | | |
| Halkn | | | | | |
| Htb | nh chur | 2 | | | |
| Htbg | ai kiếm | toán ổn định chi | ung của dầr | n | |
| Htong | nh cục l | oộ (dâm không c | o sướn tăn: | g aường ngai | ng): |
| Htongg | - | δ= | 1.66 | (cm) | |
| 6 * Chiếu a | ao sườn dầm | hs= | 94.8 | (cm) | |
| 7 => | Phải kiểm tra ổn ở | lịnh cục bộ của (| dâm | | |
| 8 Công thú | ie kiểm tra 📝 | 12 (| 2 | | |
| Q | ďσ | _ p _ [τ | < m | | |

Hinh 1.4

| Create Names | ? × |
|-----------------|--------------|
| Create names in | A CONTRACTOR |
| Top row | |
| Left column | 時に |
| Bottom row | LoX. |
| Right column | at D |
| OK Ca | ancel |

Hình 1.3

Cách thứ 2:

- Nháy chuột vào hộp *Name Box* trên dải công thức và chọn tên cần thiết từ danh sách (hoặc gõ địa chỉ của ô).

- Đây cũng là phương pháp kiểm tra xem tên đã được đặt tên cho é (hay miền) nào.

- Xoá tên.

- Chon Insert \ Name \ Define.

- Chọn tên cần xoa từ danh sách.

E Chon Delete.

- Chọn OK hoặc Close.

Chú ý: Nếu muốn lấy tên của nhóm ô này để đặt tên cho nhóm ô khác thì trước hết cần phải xoá tên đó đi.

1.2.5. Sao chép và di chuyển các tham chiếu ô

Như đã nói ở trên, khi sao chép công thức từ một ô này sang ô khác, các tham chiếu ô tuyệt đối sẽ không thay đổi, nhưng các tham chiếu ô tương đối thì thay đổi tuỳ thuộc vào vị trí của ô mà công thức được sao chép đến.

Khi di chuyển các nội dung của ô thì Excel giả thiết rằng bạn đang chỉ thay đổi bố trí chung của bảng tính mà không muốn thay đổi lôgic toán học. Vì vậy mọi tham chiếu ô trong các ô được di chuyển mà chỉ đến các ô nằm ngoài các ô được di chuyển. Chúng vẫn còn chỉ đến các ô tương tự sau khi di chuyển. Bất kỳ các ô nào mà tham chiếu đến các ô được di chuyển đều sẽ được hiệu chỉnh để chúng vẫn còn chỉ đến các ô có cùng nội dung giống như các ô mà chúng chỉ đến từ lúc đầu. Thao tác di chuyển sẽ không thay đổi bất kỳ kết quả nào của công thức, trừ khi bạn xoá dữ liệu bằng cách di chuyển các ô bên trên nó.

1.3. SỬ DỤNG CÁC TOÁN TỬ ĐỂ TÍNH TOÁN

Các toán tử là các khối được lập sẵn để tính toán. Tác động của chúng xác định xem bao nhiêu con số (hoặc chuỗi ký tự) sẽ được tổ hợp lại để tạo ra một kết quả bằng số. Excel cung cấp 3 kiểu toán tử là: toán tử tính, toán tử vān bản, toán tử lôgic. Toán tử văn bản đơn là toán tử "và", ký hiệu là &. Nó dùng để nối các chuỗi ký tự với nhau để tạo ra một chuối văn bản duy nhất. Bảng 1.1 liệt kê mọi toán tử sẵn có trong Excel.

1.3.1. Các toán tử tính

Các toán tử tính bao gồm một bộ tiêu chuẩn mà các kỹ sư đều mong đợi nó sẽ có trong bất cứ ngôn ngữ lập trình cao cấp nào.

Các toán tử tính thường dùng nhất là: cộng (+), trừ (-), nhân (*), chia (/), luỹ thừa (^), tính phần trăm (%).

| Toán tử | Mô tả | Thứ tự thực hiện |
|-----------------------|-------------------|------------------|
| Các to án tử Unary | | |
| - | Lấy dấu âm | 1 |
| % | Tính phần trăm | 2 |
| Các toán tử tính toán | | |
| ٨ | Tính luỹ thừa | 3 |
| * | Nhân | 4 |
| 1 | Chia | 4 |
| + | Cộng | 5 |
| - | Trừ | 5 |
| Toán tử văn bản | | |
| & | Và | 6 |
| Các toán tử lôgic | | |
| = | Bằng nhau | |
| < | Nhỏ hơn | |
| > | Lớn hơn | |
| <= | Nhỏ hơn hoặc bằng | |
| >= | Lớn hơn hoặc bằng | |
| <> | Không bằng nhau | |

Bảng 1.1. Các toán tử của Excel và thứ tự ưu tiên

1.3.2. Các toán tử lôgic

Các toán tử lôgic được dùng để so sánh hai giá trị bằng số hay hai chuỗi ký tự. Kết quả của toán tử là giá trị TRUE hay FALSE. Khi bạn dùng các kết quả lôgic trong các công thức toán học thì TRUE tương ứng là 1 và FALSE tương ứng là 0.

Khi Excel so sánh 2 chuỗi ký tự bằng toán tử lôgic thì nó sẽ không phân biệt chữ hoa hay chữ thường. Tuy nhiên bạn có thể dùng hàm số EXACT của Excel để so sánh các chuỗi có xét đến chữ hoa hay chữ thường.

1.3.3. Trình tự ưu tiên thực hiện các toán tử

Trình tự ưu tiên thực hiện các toán tử xác định xem phương trình sẽ được tính như thế nào. Bảng 1.1 liệt kê trình tự ưu tiên cho mỗi toán tử. Trong bất kỳ tính toán nào, các toán tử có số ưu tiên là 1 sẽ được thực hiện trước tiên, rồi đến các toán tử có số ưu tiên là 2, tiếp theo tương tự. Nếu có 2 toán tử có cùng số ưu tiên thì phép tính sẽ thực hiện ưu tiên từ trái sang phải. Bạn nên dùng thêm các dấu ngoặc nữa để quyết định thứ tự ưu tiên các toán tử cho đúng theo ý mình.

1.4. SỬ DỤNG CÁC HÀM BẠNG TÍNH

Một ưu điểm quan trọng của Excel khi thực hiện các tính toán khoa học và kỹ thuật là nó có sẵn rất nhiều hàm toán học thuộc đủ loại. Nếu thiếu những hàm này thì ngay cả các phép toán đơn giản cũng trở nèn khó khān. Ví dụ bạn đã bao giờ tính sin của một góc bằng một máy tính bấm tay thường chưa. Đó là việc khó và dễ sai sót.

Excel cung cấp 11 loại hàm bảng tính về: toán học, kỹ thuật, lôgic, luận lý, chuỗi ký tự, thống kê, ngày/ tháng, cơ sở dữ liệu, tài chính, thông tin và tra cứu. Hơn nữa bạn còn có thể dùng vô số hàm macro để lập ra các chương trình trong Excel. Nếu các hàm nội vẫn chưa đủ dùng cho bạn thì bạn có thể tự tạo thêm hàm mới của mình (hàm ngoại) bằng ngôn ngữ C hay Pascal, Fortran, Visual Basic, v.v... rồi gọi chúng từ bên trong Excel khi cần dùng.

Trong sách này sẽ nói nhiều về các hàm bảng tính mà thường được dùng trong các tính toán khoa học kỹ thuật. Để tra cứu đầy đủ về tất cả các hàm bảng tính, nên tìm cuốn sách "Tra cứu các hàm của Microsoft Excel" đã được dịch sang tiếng Việt.

1.4.1. Hàm nhập dữ liệu

Một hàm Excel thường có chứa tên hàm, dấu mở ngoặc, một vài đối số cách nhau bởi các dấu phẩy rồi đến các dấu đóng ngoặc. Các đối số của hàm có thể là con số, chuỗi ký tự, tham chiếu ô hoặc các hàm khác (nhưng chỉ lồng nhau được nhiều nhất 7 cấp hàm). Nếu đối số của hàm không nằm trong phạm vi được hàm chấp nhận thì hàm sẽ cho giá trị là #NUM! trong ô. Nếu đối số không đúng kiểu mà hàm yêu cầu thì sẽ có kết quả là #VALUE!.

Trong sách này các tên hằm được viết bằng chữ hoa và các đối số được viết theo kiểu chữ Italic để bạn đọc dễ phân biệt. Nhưng bạn gõ kiễu chữ nào thì Excel cũng hiểu cả.

Cần nhớ rằng một số hàn ở dạng các tệp Macro bổ sung (add-in macro files), chúng cần được gắn vào Excel từ trước lúc bạn muốn dùng. Cách làm như sau: bạn dùng lệnh Add-Ins của thực đơn Optiom để gọi Add-In Manager. Nhắp chuột vào núm Add trong Add-In Manager để tìm và gắn một tệp dạng Add-Ins. Sau khi bạn đã gắn kết các tệp Macro bổ sung đó thì Exceil sẽ thực hiện mọi công thức của bạn có dùng đến những hàm này.

1.4.2. Hàm mảng

Một hàm thông thường sẽ cho kết quả là trị số riêng lẻ. Tuy nhiên Excel cung cấp các hàm mảng để cho kết quả là nhiều giá trị đồng thời. Ví dụ hàm nghịch đảo ma trận là MINVERSE cho kết quả là một mảng.

Hàm mảng phải được nhập vào dãy ô mà dãy này cần đủ lớn để chứa tất cả các trị số kết quả. Ví dụ nếu bạn diùng hàm MINVERSE để nghịch đảo một mảng kích thước 3×3 thì kết quả cũng sẽ là mảng 3×3 , do đó bạn phải đặt hàm trong vùng ô có kích thước 3×3 .

Để chèn hàm mảng vào một vùng ô thì hãy chọn các ô rồi gõ hàm mảng và các đối số của nó vào ô trên cùng bên trái. Sau đó nhấn phím Ctrl và phím Shift đồng thời nhấp chuột trên ô đánh dấu rồi bấm Enter. Hàm mảng được bao bởi dấu ngoặc tròn () sẽ được đặt tại mọi chỗ trong vùng ô đã chọn. Bạn không phải gõ dấu ngoặc tròn vì Excel sẽ chèn chúng vào để đánh dấu sự nhắp vào như là của một hàm mảng.

Thực ra, mọi công thức áp dụng cho mảng hay cho một trị số đều theo cách thức giống nhau. Ví dụ, công thức sau đây tính tổng của các trị số trong mảng hình vuông rồi cho kết quả là một số như sau:

= SUM ((A7:A15-\$B\$1)^2)

Nếu bạn gõ công thức này rồi nhấn phím Ctrl và Shift trong khi nhấp chuột vào ô kiểm tra (hoặc nhấn Enter) thì trước tiên Excel sẽ trừ đi trị số trong ô BI khỏi tất cả các trị số trong các ô từ A7 đến A15, rồi lấy bình phương của từng kết quả, sau đó cộng chúng lại. Nếu nhập công thức này trong lúc bạn không nhấn giữ các phím Ctrl và phím Shift thì Excel sẽ chỉ dùng 1 trong các trị số của các ô từ A7 đến A15 (lấy ô nào ở cùng dòng với ô có chứa công thức).

1.4.3. Hàm toán học

Các hàm toán học lấy các đối số là những dữ liệu số rồi xử lý và tạo ra một kết quả bằng số. Có 4 dạng hàm toán học chủ yếu là: toán cơ bản, lôgarit, lượng giác, ma trận. Bảng 1.2 liệt kê các hàm đó.

| Hàm Số | Kết quả tính toán | | | |
|---------------------|--|--|--|--|
| Các hàm toán học cơ | Các hàm toán học cơ bản | | | |
| ABS() | Tính giá trị tuyệt đối | | | |
| BASE() | Chuyển một số hệ thập phân sang hệ cơ số khác | | | |
| CEILING() | Làm tròn đến số nguyên gần nhất | | | |
| COMBIN () | Tính số tổ hợp cho một số đối tượng tuỳ theo con số trong mỗi lần chọn | | | |
| COUNTBLANK () | Đếm số lượng ô trống trong một hàng | | | |
| COUNTIF() | Đếm số ô phù hợp với tiêu chuẩn nào đó định sẵn | | | |
| EVEN() | Làm tròn một số tới số nguyên chẵn gẩn nhất | | | |
| FACT () | Tính giai thừa của một số | | | |
| FACTDOUBLE () | Tính giai thừa bội của một số cho trước | | | |
| FLOOR () | Làm tròn dưới tới một số nguyên gần nhất hay theo con số định sắn | | | |
| GCD() | Tính ước số chung lớn nhất của 2 hay nhiều số | | | |

Bảng 1.2. Các hàm toán học của Excel

| Hàm số | Kết quả tính toán |
|--------------------|---|
| LCM() | Tính bhội số chung nhỏ nhất của 2 hay nhiều số |
| INT() | Làm 1 trờn dưới tới một số nguyên |
| MOD() | Cho phầm dư của phép toán chia 2 số |
| MROUND() | Làm tròm một số tới một số cho trước |
| MULTINOMINAL | Tínhi tỷ liệ giữa tổng của các giai thừa và tích các giai thừa |
| ODD() | Làma tròm trên tối một số nguyên lẻ gần nhất |
| POWER () | Tính lũy thừa của một số |
| PRODUCT() | Nhâm một hoạt số |
| QUOTIENT () | Cho phẩm nguyên của một phép tính chia |
| RAND() | Cho một số ngẫu nhiên trong khoảng từ 0 đến 1 |
| RANDBETWEEN () | Cho một số ngẫu nhiên trong khoảng xác định trước |
| ROUND() | Làm trrồm một số theo số đã cho trước |
| SERIESUM () | Tính tổng của một cấp số |
| SIGN () | Cho dấtu của một số, trả về 1 nếu số đó dương, 0 nếu số đó âm |
| SQRT() | Tính bình phương của một số |
| SQRTP() | Tính bình phương nhiều lần của một số |
| SUM() | Còng ciác: số trong một danh sách cho trước |
| SUMIF() | Tính tổ ng các số mà thoả mãn cùng một tiêu chuẩn nào đó định sẫn |
| SUMPRODUCT () | Tính tổng các tích của các phần từ ma trận |
| SUMSQ() | Tínhi tổ ng bình phương của các số trong một danh sách cho trước |
| SUMX2MY2() | Tính tổởngg các hiệu số của bình phương các phần tử trong 2 ma trận |
| SUMX2PY2() | Tính tổ ng các tổng số của bình phương các phần tử trong 2 ma trận |
| SUMXMY2() | Tính tổởng các bình phương của các hiệu số giữa các phần tử trong hai ma trận |
| TRUNC() | Loại bở bớt các số tuỳ theo số lượng chữ số muốn giữ lại của một số |
| Các hàm lôgarit | |
| EXP() | Tính hànn số mũ e của một số |
| EXP(1) | Cho Itri sco c.ua e (= 2,718284590) |
| LN() | Tính lô)gzaritt tự nhiên của một số |
| LOG() | Tinhi lôjgariit cơ số nào đó cho trước của một số |
| LOG10() | Tính lôjgaariit cơ số 10 của một số |
| Các hàm lượng giác | |
| COS() | Tíinh cosiin của một số |
| SIN() | Tiinh siin cuia một số |
| TAN() | Tíinh taangg cula một số |

Bảng 1.2 (tiếp)

| Hàm số | Kết quả tính toán |
|----------------------|--|
| Các hàm lượng giác n | gược |
| ACOS() | Tính arccos của một số |
| ASIN() | Tính arcsin của một số |
| ATAN() | Tính arctg của một số (từ - π đến + $\pi/2$) |
| ATAN2 (x, y) | Tính arctg của một số (từ $-\pi$ đến + π); x, y là toạ độ của điểm tính |
| Các hàm hyperbolic | |
| COSH () | Tính cosin hypebolic của một số |
| SINH() | Tính sin hypebpolic của một số |
| TANH() | Tính tang hypebolic của một số |
| Các hàm hyperbolic n | gược |
| ACOSH() | Tính cosin hypebolic ngược của một số |
| ASINH() | Tính sin hypebolic ngược của một số |
| ATANH() | Tính tang hypebolic ngược của một số |
| PI | Cho giá trị của số π (= 3,1415926535898) |
| Các hàm đổi đơn vị đ | o góc |
| DEGREES() | Đổi từ đơn vị là radian sang đơn vị là độ |
| RADIAN () | Đổi từ đơn vị là độ sang đơn vị là radian |
| Các hàm ma trận | |
| MDETERM () | Tính định thức của ma trận |
| MINVERSE () | Nghịch đảo ma trận |
| MMULT() | Nhân 2 ma trận với nhau |
| TRANSPOSE () | Lập ma trận chuyển trí (đổi hàng thành cột, dồi cột thành hàng) |

1.4.4. Hàm kỹ thuật

Các hàm kỹ thuật có sẵn trong Excel là:

- Các hàm Bessel.
- Các hàm chuyển đổi cơ số.
- Các hàm số góc ngược.
- Các hàm lỗi.
- Các hàm so sánh số.
- Các hàm số phức.

Cân lưu ý rằng số phức được giữ trong máy tính đưới dạng chuỗi ký tự dạng (x + yj) trong đó x là phần thực còn y là phần ảo. Bảng 1.3 liệt kê các hàm kỹ thuật của Excel.

| Hàm số | Kết quả tính toán | |
|---------------------------------|---|--|
| Các hàm Bessel | * <u></u> | |
| BESSELJ() | Hàm Bessel $J_n(x)$ | |
| BESSELI() | Hàm Bessel cải biên J _m (x) | |
| BESSELK () | Hàm Bessel cải biên K _n (x) | |
| BESSELY() | Hàm Bessel Weber Y _m (x) | |
| Các hàm chuyển đổi cơ số đếm | | |
| BIN2DEC() | Chuyển đổi từ số đếm ở hệ nhị phân sang hệ cơ số 10 | |
| BIN2HEX() | Chuyển đổi từ số đếm ở hệ nhị phân sang hệ đếm cơ số 16 | |
| BIN2OCT() | Chuyển đổi từ số đếm ở hệ nhị phân sang chuỗi octan | |
| CONVERT() | Chuyển đổi một số từ một đơn vị đo này sang đơn vị đo khác | |
| DEC2BIN() | Chuyển đổi từ số nguyên thập phân sang chuỗi nhị phân | |
| DEC2HEX() | Chuyển đổi từ số nguyên thập phân sang chuỗi cơ số 16 | |
| DEC2OCT() | Chuyển đổi từ số nguyên thập phân sang chuỗi octan | |
| FACTDOUBLE () | Tính giai thừa gấp đôi | |
| HEX2BIN() | Chuyển đổi từ cơ số 16 sang số nhị phân | |
| HEX2DEC() | Chuyển đổi từ cơ số 16 sang số thập phân | |
| HEX2OCT() | Chuyển đổi từ cơ số 16 sang chuỗi octan | |
| OCT2BIN() | Chuyển đổi từ số octan sang chuỗi nhị phân | |
| OCT2DEC() | Chuyển đổi từ số octan sang số thập phân | |
| OCT2HEX() | Chuyển đổi từ số octan sang chuỗi cơ số 16 | |
| Các hàm sai số | | |
| ERF() | Hàm sai số | |
| ERFC() | Hàm sai số có bình luận | |
| Các hàm so sánh giá trị bằng số | | |
| DELTA() | Hàm Delta, trả về giá trị 1 nếu cả hai số giống nhau, hoặc | |
| | trả về giá trị 0 nếu chúng khác nhau | |
| GESTEP() | Hàm bước, trả về giá trị 1 số là lớn hơn bước, hoặc trả về | |
| | giá trị 0 nếu số là nhỏ hơn hoặc bảng bước | |
| Các hàm sô phức | | |
| COMLEX () | Chuyển đổi từ 2 hệ số thành một số phức có dạng (x + yj) | |
| IMABS() | Trị số tuyệt đối của số phức trong một chuỗi | |
| IMAGINARY () | Hệ số ảo y của số phức trong một chuỗi | |
| IMARGUMENT() | Góc, tính bằng radian, trong mặt phẳng phức của số phức trong chuỗi | |

Bảng 1.3. Các hàm kỹ thuật của Excel

Bång 1.3 (nép)

| Hàm số | Kết quả tính toán |
|-----------------|--|
| IMCONJUGATE () | Liên kết phức của số phức trong chuỗi |
| IMCOS () | Cosin của số phức trong chuỗi |
| IMDIV () | Thương số của 2 số phức trong chuổi |
| IMEXP() | Luỹ thừa có số e của số phức trong chuỗi |
| IMLN() | Lôgarit tự nhiên của số phức trong chuối |
| IMLOG2() | Lôgarit cơ số 2 của số phức trong chuồi |
| IMLOG10() | Lôgarit cơ số 10 của số phức trong chuỗi |
| IMPOWER () | Luỹ thừa số nguyên của số phức trong chuỗi |
| IMPRODUCT () | Tích của 2 số phức trong chuỗi |
| IMREAL () | Hệ số thực của số phức trong chuỗi |
| IMSIN () | Sin của số phức trong chuỗi |
| IMSQRT () | Bình phương của số phức trong chuỗi |
| IMSUB() | Hiệu của 2 số phức trong chuỗi |
| IMSUM () | Tổng của 2 hay nhiều số phức trong chuỗi |

1.4.5. Các hàm lôgic

Các hàm lôgic sẽ trả về giá trị bằng "TRUE" hoặc "FALSE". Các giá trị này tương ứng với số 1 hoặc số 0. Ví dụ TRUE $\times X = X$ và FALSE $\times X = 0$.

Công dụng thông thường nhất của hàm IF là cảnh báo để tránh các phép tính số học không hợp lệ (chẳng hạn như phép chia cho số 0) và để lựa chọn một trong các tình huống tính toán. Ví dụ để tính toán giá trị của sin(x)/x đối với mọi giá trị của x, thì có thể dùng công thức:

IF $(x = 0, 1, \sin(x)/x)$

Nếu x bằng 0 thì hàm số trả về giá trị đúng (hoặc 1), nếu x khác 0 thì hàm số sẽ tính toán rồi cho giá trị của sin(x)/x. Nếu chúng ta không dùng hàm IF ở đây thì sẽ gặp lỗi chia cho số 0 tức là trong ô sẽ hiện lên #DIV/ 0! khi mà x = 0, cho dù lúc đó đáp số đúng phải là sin (0)/ 0 = 1.

Trong số các hàm luận lý còn có 3 toán tử Boolean. Đó là AND (A. B) (mang nghĩa là A và B), toán tử OR (A, B) mang nghĩa là A hoặc B và toán tử NOT (A. B) mang nghĩa là không là A và không là B. Chúng kết hợp các giá trị lôgic tuỳ theo các quy tắc của đại số Boolean để tạo ra một kết quả lôgic. Các toán tử Boolean thường được dùng như các toán tử 2 chiều. Tuy nhiên Excel coi chúng như là các hàm số.

Một số toán tử Boolean khác ít được dùng là toán tử XOR (A, B), mang nghĩa là không là A hoặc không là B, toán tử EQV mang nghĩa là tương đương lôgic, toán tử IMP

mang nghĩa là bao hàm lôgic. Tuy nhiên chúng ta có thể kết hợp từ 3 hàm lôgic cơ bản để thay cho việc phải dùng 3 hàm bổ sung nói trên. Ví dụ như:

- đối với toán tử XOR có thể thay bằng công thức:

= OR (AND(A, NOT (B)), AND (B, NOT(A)))

- đối với toán tử EQV có thể thay bằng công thức:

= OR (AND (NOT (A), NOT (B)), AND (B, A))

- đối với toán tử IMP có thể thay bằng công thức:

= OR (NOT (OR (A,B)), B).

| Bång 1.4. Cád | : hàm | lôgic | của | Excel |
|---------------|-------|-------|-----|-------|
|---------------|-------|-------|-----|-------|

| Hàm số | Kết quả tính toán | | |
|--|---------------------------------------|--|--|
| Các hàm giá trị lôgic | | | |
| TRUE() | True hoặc 1 | | |
| FALSE() | False hoặc 0 | | |
| Các hàm thử nghiệm trị lôgic | | | |
| IR Chọn 1 trong 2 trị số tuỳ theo giá trị lô | | | |
| Các hàm (toán tử Boolean) | | | |
| AND (A, B) | Ý nghĩa là A và B | | |
| NOT (A, B) | Ý nghĩa là không là A cũng không là B | | |
| OR (A, B) | Ý nghĩa là A hoặc B | | |

1.4.6. Hàm văn bản

Một chuỗi gồm các ký tự văn bản giống như một câu văn. Các hàm văn bản có khả năng sửa chữa thay đổi hay tạo ra một chuỗi ký tự. Bảng sau đây liệt kê các hàm ký tự của Excel.

Các hàm DOLLAR (số, chữ số) và FIXED (con số, chữ số, không dấu phẩy) có khả năng làm tròn một giá trị số phù hợp với số chữ số đã định trước rồi chuyển chúng thành một chuỗi các ký tự. Nếu tham số "không dấu phẩy" lấy giá trị lôgic là TRUE thì hàm FIXED sẽ không chèn dấu phẩy vào chuỗi kết quả. Sự khác nhau giữa hai hàm này là ở chỗ hàm DOLLAR sẽ tạo ra một số dưới dạng thông thường. Cũng giống như hàm ROUND những số âm sẽ được làm tròn về phía bên trái của dấu phẩy thập phân.

Hàm LEN (văn bản) định nghĩa chiều dài của chuỗi văn bản.

Các hàm MID (văn bản, bắt đầu, số), hàm LEFT (văn bản, số) và hàm RIGHT (văn bản, số) sẽ khai triển các ký tự từ bên trong của chuỗi hoặc từ hai đầu chuỗi. Các ký tự được đánh số theo quy tắc: ký tự đầu tiên là 1, ký tự thứ hai là 2.

Hàm SUBSTITUTE (văn bản, cũ, mới, số) và hàm REPLACE (văn bản, mở đầu, số, mới) thay thế một đoạn bên trong chuỗi.

Hàm SEARCH (văn bản 1, văn bản 2, mở đầu) và hàm FIND (văn bản 1, văn bản 2, mở đầu) sẽ xác định vị trí của một đoạn bên trong chuỗi.

| Hàm số | Kết quả tính toán | | |
|-------------------------------------|---|--|--|
| Các hàm chuyển đối số thành văn bản | | | |
| DOLLAR () | Đổi một số thành một văn bản dạng thông thường | | |
| FIXED() | Đổi một số thành một văn bản sau khi làm tròn với số chữ số định trước | | |
| TEXT() | Đôi một giá trị thành văn bản | | |
| VALUE() | Đổi một số thành chuỗi có dạng định trước | | |
| Các hàm điều khiến văn bản | | | |
| CLEAN() | Lấy bớt đi các ký tự không thể in được ra khỏi văn bản | | |
| CONCATENATE () | Kết hợp hai hay nhiều chuỗi với nhau | | |
| FIND() | Tìm một đoạn trong chuỗi | | |
| LEFT() | Khai triển các ký tự từ bên trái | | |
| LEN() | Cho biết số ký tự trong chuỗi | | |
| Các hàm chuyển đổi số thành văn bản | | | |
| DOLLAR () | Số đổi thành văn bản đạng thông thường | | |
| LOWER () | Chuyển văn bản thành chữ thường | | |
| MID() | Khai triển một đoạn của chuỗi | | |
| PROPER () | Viết hoa chữ đầu tiên mỗi từ trong câu | | |
| REPLACE () | Thay thế một đoạn câu | | |
| RIGHT() | Khai triển các ký tự từ bên phải | | |
| SEARCH() | Tìm vị trí một đoạn câu | | |
| SUBTITUE () | Thay thế nhiều đoạn câu | | |
| TRIM() | Cắt bỏ khoảng trống ở đầu | | |
| UPPER () | Chuyển mọi chữ trong văn bản thành chữ hoa | | |
| Các hàm tạo ra chuối | | | |
| CHAR () | Tạo ra các ký tự theo mã ASCII | | |
| CODE () | Cho biết mã ASCII của ký tự | | |
| REPT() | Lặp lại một chuỗi | | |
| Hàm nén chuỗi | 171 | | |
| EXACT() | Hàm nén chuỗi | | |

3

| Bảng 1.5. Các hàm | văn bản | của Exc | el |
|-------------------|---------|---------|----|
|-------------------|---------|---------|----|

1.4.7. Hàm ngày tháng và hàm thời gian

Các hàm ngày tháng thực hiện tính toán hoặc chuyển đổi ngày tháng hoặc ghép ngày tháng với thời gian. Bạn đọc nên lưu ý rằng trong Excel đã định nghĩa: số ngày danh định là số ngày kể từ hôm 1-1-1900 đến ngày đang xét. Mọi số ngày được nhập từ bàn phím bằng tay hoặc được tính ra bằng công thức sẽ được lưu trong máy tính dưới dạng trị số ngày danh định. Nếu muốn xem một ngày thực mà con số đại diện thì phải định dạng ô chứa ngày đó là dạng DATE.

Các giá trị thời gian đếm theo đồng hồ cũng được lưu trữ dưới dạng số thời gian danh định là một phần của ngày theo quy tắc đếm như sau: nửa đêm là thời điểm 0, giữa trưa là thời điểm 0,5. Để thể hiện thời gian trong ô của bảng tính thì phải định dạng ô là dạng TIME.

Bởi vì ngày và thời gian đều được thể hiện theo đơn vị đo là ngày nên chúng ta có thể kết hợp chúng theo cách rất đơn giản là cộng chúng lại với nhau. Để tìm hiệu số giữa 2 ngày hoặc 2 thời điểm nào đó thì chỉ cần trừ chúng cho nhau. Kết quả sẽ tính theo đơn vị ngày hoặc một phần của ngày. Bảng 1.6 liệt kê các hàm ngày tháng và hàm thời gian của Excel. Các hàm NOW và TODAY sẽ cho giá trị thời gian và ngày tháng hiện hành.

1.4.7.1. Các hàm ngày tháng

Hàm DATE (*năm*, *tháng*, *ngày*) chuyển đổi một ngày nào đó thành số ngày danh định. Hàm này dùng để tạo ra số ngày tháng căn cứ vào dữ liệu năm tháng ngày đã cho trước.

Hàm DATEVALUE (văn bản) chuyển đổi một đoạn văn bản của ngày tháng thành số ngày danh định.

Các hàm YEAR (số), hàm MONTH (số) và hàm DAY (số) cho biết các con số của năm, hoặc của tháng trong năm, hoặc của ngày trong tháng khi mà chúng ta cho trước con số. Hàm này có chức năng ngược với hàm DATE.

Hàm WEEKDAY (số) cho biết giá trị bằng số của ngày trong tuần. Ví dụ: ngày chủ nhật là 1, thứ bảy là 7.

1.4.7.2. Các hàm thời gian

Hàm TIME (giờ, phút, giây) cho biết một thời điểm nào đó là bao nhiêu phần của một ngày. Các tham số trong dấu ngoặc phải là số nguyên, nếu bạn nhập một số thực thì sẽ bị máy tính lờ đi phần sau dấu phẩy thập phân.

Hàm TIMEVALUE (văn bản) chuyển đổi thời gian lưu trữ trong chuỗi thành số ngày danh định.

Hàm HOUR (số), hàm MINUTE (số) và hàm SECOND (số) là hàm ngược của hàm TIME (số). Chúng cho biết về giờ, phút và giây tuỳ theo thời gian được lưu trữ trong tham số "số". Lưu ý rằng Excel luôn làm tròn mọi số đến số thứ hai gần nhất.

| Hàm số | Kết quả tính toán |
|-----------------------------|--|
| Các hàm ngày tháng hiện tại | |
| NOW() | Số ngày danh định của ngày tháng hiện tại |
| TODAY() | Số ngày danh định của ngày hôm nay |
| Các hàrn ngày tháng | |
| DATE () | Số ngày danh định của ngày tháng hiện tại |
| DATEVALUE () | Chuyển đổi số ngày dạng văn bản thành dạng danh định |
| DAY() | Cho biết ngày trong tháng tuỳ theo số ngày danh định |
| DAYS360() | Số lượng ngày đếm giữa 2 ngày nào đó cho trước |
| EDATE() | Số ngày danh định của ngày mà ở trước hoặc sau ngày nào đó một số tháng cho trước |
| EDMONTH() | Số ngày danh định của ngày gần nhất của tháng mà ở trước hoặc sau ngày nào đó một số tháng cho trước |
| MONTH() | Tháng của năm tuỳ theo số ngày danh định |
| NETWORKDAYS() | Số ngày làm việc đếm giữa 2 ngày nào đó cho trước |
| WEEKDAY() | Ngày của tuần lễ tuỳ theo số ngày danh định. Chủ nhật là 1 và thứ bẩy là 7 |
| WEEKNUM() | Số tuần lễ trong năm của ngày định trước |
| WORKDAY() | Ngàv tháng của một ngày làm việc nào đó trong tương lai |
| YEAR () | Số năm tuỳ theo số ngày danh định |
| YEARFRAC() | Số phần của năm là hiệu số giữa 2 ngày nào đó định trước |
| Các hàm thời gian | |
| HOUR () | Giờ của ngày tuỳ theo số ngày danh định |
| MINUTE() | Phút của giờ tuỳ theo số ngày danh định |
| SECOND() | Giây của phút tuỳ theo số ngày danh định |
| TIME() | Số ngày danh định của giờ, phút và gi a y |
| TIMEVALUE() | Đổi thời gian từ dạng văn bản sang dạng số ngày danh định |

Bảng 1.6. Các hàm ngày tháng và hàm thời gian

1.4.8. Các hàm thống kê

Các hàm thống kê nói chung được áp dụng cho các phép tính thống kê. Ví dụ như các phép tính tổng và kỳ vọng, phương sai và cả các phép tính hàm xấp xỉ thống kê theo các số liệu thực nghiệm. Trong bảng 1.7 liệt kê các hàm thống kê của Excel.

1.4.8.1. Các hàm thống kê cơ bản

Các hàm AVERAGE (*danh sách số*) và hàm COUNT (*danh sách số*) thao tác trên danh sách các số và cho biết kỳ vọng, xắp xếp thứ tự to nhỏ cho các số trong danh sách số.

Các hàm MIN (*danh sách só*) và MAX (*danh sách sô*) cho biết các trị số nhỏ nhất và trị số lớn nhất trong danh sách số. Những ô trong bảng tính để trống hoặc chứa các ký tự, các giá trị lôgic sẽ được lờ đi không xét.

Giá trị trung bình hình học được tính bằng hàm GEOMEAN (*danh sách số*). Giá trị trung bình điều hoà được tính bằng hàm HARMEAN (*danh sách số*). Trị số kỳ vọng của phân phối được tính bằng hàm MEDIAN (*danh sách số*).

Hàm STDEV (*danh sách số*) tính độ lệch chuẩn mẫu của các giá trị trong danh sách số. Nó chính là ước lượng tốt nhất của độ lệch chuẩn của tập hợp số và tính theo công thức:



Trong đó: $\overline{\mathbf{x}}$ = trung bình của n giá trị \mathbf{x}_i .

Nếu danh sách số bao gồm toàn bộ các số của tập hợp chứ không phải chỉ gồm một số mẫu lây ra từ tập hợp số cho trước thì dùng hàm STDEVP để tính theo công thức:



Hàm phương sai VAR (*danh sách số*) chỉ là bình phương của độ lệch chuẩn. Nó chính là phương sai mẫu của tập hợp số. Để tính phương sai của cả tập hợp số khi danh sách số bao gồm toàn bộ tập hợp số thì dùng hàm VARP (*danh sách số*), đó là bình phương của độ lệch chuẩn của tập hợp số.

| Hàm số | Kết quả tính toán | | |
|-------------------------|---|--|--|
| Các hàm thống kê cơ bản | | | |
| AVERAGE () | Số trung bình công của danh sách số | | |
| COUNT() | Số lượng giá trị số trong danh sách số | | |
| COUNTA () | Số lượng các ô không trống trong danh sách | | |
| DEVSQ () | Tổng bình phương các phương sai | | |
| GEOMEAN () | Trung bình hình học của danh sách số | | |
| HARMEAN() | Trung bình điều hoà của danh sách số | | |
| MAX () | Số lớm nhất trong danh sách số | | |
| MEDIAN() | Số giữa danh sách sau khi đã xắp xếp các số từ to đến nhỏ | | |
| MIN() | Số nhỏ nhất trong danh sách số | | |

Bång 1.7. Các hàm thống kê của Excel

| Hàm số | Kết quả tính toán | | | |
|---------------------------|--|--|--|--|
| STDEV () | Độ lệch chuẩn của các số trong danh sách | | | |
| STDEVP() | Độ lệch chuẩn của các số, giả thiết danh sách là toàn bộ tập hợp số | | | |
| VAR() | Phương sai của các số trong danh sách số | | | |
| VARP() | Phương sai của các số trong danh sách số, giả thiết danh sách là toàn bộ tập hợp số | | | |
| Các hàm thống kê cao cấp | | | | |
| AVEDEV () | Độ lệch trung bình của các điểm trong danh sách so với số trung bình | | | |
| CORREL () | Hệ số tương quan giữa 2 danh sách . | | | |
| CONFIDENCE () | Khoảng tin cậy tập hợp số | | | |
| COVAR () | Hiệp phương sai giữa 2 danh sách | | | |
| FISHER () | Biến đổi Fisher của một số | | | |
| FISHERINV () | Nghịch đảo của biến đổi Fisher của một số | | | |
| FREQUENCY () | Phân phối tần số của giá trị trong danh sách | | | |
| KURT() | Kurtosis của các giá trị trong danh sách | | | |
| LARGE() | Trị số rộng nhất thứ k trong danh sách | | | |
| MODE() | Trị số chung nhất trong danh sách | | | |
| PEARSON () | Hệ số tương quan mômen tích Pearson của 2 mảng | | | |
| PERCENTILE () | Giá trị của phần trăm thứ k trong danh sách | | | |
| PERCENTRANK () | Thứ hạng phần trăm của số trong danh sách | | | |
| PERMUT() | Số các hoán vị của các đối tượng đã chọn đồng thời từ một số định sẵn các đối tượng | | | |
| PROB() | Xác suất để các giá trị trong danh sách là nằm giữa các số nhỏ hơn và số lớn hơn nào đó | | | |
| QUARTILE () | Các giới hạn bậc 4 từ danh sách | | | |
| RANK () | Hạng của số trong danh sách | | | |
| SMALL() | Trị số nhỏ nhất thứ k trong danh sách | | | |
| STANDARDIZE () | Chuẩn hoá một số với kỳ vọng và độ lệch chuẩn | | | |
| TRIMMEAN () | Số trung bình của danh sách với các điểm phần trăm, lấy ra từ các đầu của phân phối | | | |
| Các hàm xấp xỉ đường cong | | | | |
| FORECAST () | Ngoại suy giá trị với xấp xỉ dạng tuyến tính | | | |
| GROWTH() | Ngoại suy giá trị với xấp xỉ dạng hàm mũ e | | | |
| LOGEST () | Các tham số của một xấp xỉ dạng hàm mũ e | | | |
| RSQ() | Độ phù hợp của xấp xỉ đường cong, trị số r ² cho đường hồi quy tuyến tính | | | |

| Hàm số | Kết quả tính toán | | |
|-----------------------------|---|--|--|
| SLOPE () | Độ nghiêng của đường hồi quy tuyến tính | | |
| STEYX() | Sai số chuẩn của trị số y dự đoán đối với mỗi x trên đường hồi quy tuyến tính | | |
| TREND() | Danh sách các trị số dự đoán từ đường xấp xỉ tuyến tính | | |
| Các hàm phân phối | | | |
| BETADIST () | Hàm mật độ xác suất Beta tích luỹ | | |
| BETAINV () | Nghịch đảo của hàm mật độ xác suất Beta tích luỹ | | |
| BINOMDIST () | Số hạng riêng trong phân phối nhị thức | | |
| CHIINV () | Hàm ngược của hàm phân phối χ bình phương | | |
| EXPONDIST () | Phân phối luỹ thừa | | |
| FDIST () | Phân phối xác suất F | | |
| FINV() | Hàm ngược của hàm phân phối xác suất F | | |
| GAMMADIST () | Phân phối Gamma | | |
| GAMMAINV () | Hàm ngược của phân phối tích lũy Gamma | | |
| GAMMALN() | Lôgarit tự nhiên của hàm Gamma | | |
| HYPGEOMDIST () | Phân phối siêu hình học | | |
| LOGINV () | Hàm ngược của phân phối log chuẩn | | |
| LOGNORMDIST () | Phân phối log chuẩn | | |
| NEGBINOMDIST () | Phân phối nhị thức âm | | |
| NORMDIST () | Phân phối tích luỹ chuẩn | | |
| NORMINV () | Hàm ngược của phân phối tích luỹ chuẩn | | |
| NORMSDIST () | Phân phối tích luỹ chuẩn tiêu chuẩn | | |
| NORMSINV () | Hàm ngược của phân phối tích luỹ chuẩn | | |
| POISSON () | Phân phối xác suất Poisson | | |
| SKEW() | Độ nghiêng của phân phối đại diện bởi các số trong danh sách | | |
| TDIST() | Phân phối T- Student | | |
| TINV() | Hàm ngược của phân phối T- Student | | |
| WEIBULL() | Phân phối Weibull | | |
| Các hàm thứ nghiệm dấu hiệu | | | |
| CHITEST () | Thử nghiêm χ hình phương về tính độc lập của 2 phân phối | | |
| CRITBINOM () | Giá trị nhỏ r.nất mà với nó thì phân phối nhị thức tích luỹ nhỏ hơn hay bằng giá trị tới hạn | | |
| FITEST () | Thử nghiệm F trên 2 phân phối | | |
| TTEST () | Thử nghiệm T- Student để kiểm chứng hệ số | | |
| ZTEST () | Thử nghiệm Z | | |

1.4.8.2. Các hàm thống kê cao cấp

EXCEL cung cấp rất nhiều hàm thống kê cao cấp để thử nghiệm về mối quan hệ giữa các danh sách số với nhau. Ví dụ, hàm CORREL (*danh sách số thứ 1*, *danh sách số thứ 2*) sẽ tính toán mối tương quan giữa 2 danh sách số đó, còn hàm COVAR (*danh sách số thứ 2*) sẽ tính ra hiệp phương sai.

Một số hàm thao tác với danh sách số đã được sắp xếp theo thứ tự. Ví dụ hàm LARGE (*danh sách số*, k) và hàm SMALL (*danh sách số*, k) sẽ chọn ra các giá trị từ đầu và cuối của danh sách. Hàm PERCENTRANK (*danh sách số*, số, dấu) sẽ cho biết thứ tự xếp hạng của giá trị mới.

1.4.8.3. Các hàm ngoại suy xấp xỉ đường cong

Excel có 2 dạng hàm ngoại suy là dạng tuyến tính và dạng hàm số mũ.

Các hàm LINEST (yarray, xarray), hàm TREND (yarray, xarray, xlist) và hàm FORECAST (x, yarray, xarray) sẽ tính toán đường cong xấp xỉ bình phương tối thiểu đối với dữ liệu về x và y trong 2 danh sách số là xarray và yarray.

Hàn LINEST (yarray, xarray) trả về một mảng nằm ngang gồm 2 phần tử chứa độ dốc (n), số chặn (b) của một đường thẳng theo phương trình y = mx + b.

Cá: hàm TREND (yarray, xarray, xlist) và hàm FORECAST (x, yarray, xarray) sẽ trả về các giá trị dự báo của y mà nhận được bằng cách chèn các giá trị x trong xlist hoặc trong xarray vào trong phương trình của đường thẳng. Hàm TREND trả về mảng các giá trị y, còn hàm FORECAST chỉ trả về 1 giá trị. Trong cả 2 hàm đó, nếu xarray không được nêu ra thì sẽ giả thiết rằng nó bằng danh sách các giá trị: 1, 2, 3,...

Đế ngoại suy theo xấp xỉ hàm mũ chúng ta dùng hàm LOGEST (yarray, xarray) và hàm GROWTH (yarray, xarray, xlist). Cả 2 hàm này thao tác tương tự như các hàm nội suy tuyến tính, chỉ khác là chúng tính theo hàm ngoại suy cong: $y = b.m^{x}$.

Các hệ số ngoại suy đường cong được tính toán bằng cách lấy lôgarit tự nhiên của dữ liệu yarray, khi áp dụng hàm LINEST rồi lấy luỹ thừa của kết quả sau đó xét đến m và b như siu:

EXP (LINEST (LN (yarray), xarray))

Công thức này phải được chèn vào 2 ô như một hàm mảng. Nó sẽ cho kết quả giống như lhi dùng hàm GROWTH.

Tưy nhiên nếu khoảng thay đổi của các giá trị trong *yarray* quá lớn (bao gồm từ giá trị tonhất và trị số nhỏ nhất) thì đường cong sẽ bị uốn vẹo đi. Các giá trị nhỏ hơn sẽ có ảnh lưởng lớn hơn (nghĩa là gần sát với đường cong hơn) so với các trí số lớn hơn. Nguyên nhân là do đường cong xấp xỉ gần với lôgarit của dữ liệu y hơn là gần với bản thân lữ liệu y.

Phương pháp bình phương tối thiểu có tác dụng giảm đến tối thiểu các sai số ngoại suy (độ chênh lệch giữa dữ liệu và đường cong ngoại suy). Trong trường hợp này, công thức cực tiểu hoá sai số đối với logarit của dữ liệu y chứ không phải đối với chính dữ liệu y. Mặt khác, sai số phần trăm sẽ là hằng số thô của mọi giá trị của dữ liệu y.

1.4.8.4. Các hàm thử phân phíôi và định dấu

Các hàm phân phối của Excel tạo ra các giá trị cho hầu hết các phân phối thông dụng để thử nghiệm dấu hiệu của các giái trị cũng như để dự báo về thời gian sống (tuổi thọ) hoặc mức độ hư hỏng của các siản phẩm công nghiệp. Các phân phối bao gồm: phân phối nhị thức (BINOMDIST), phân phối χ^2 (CHIDIST), phân phối F (FDIST), phân phối Gamma (GAMADIST), phân phối siêu hình học (HYPGEOMDIST), phân phối chuẩn (NORMSDIST), phân phối Poisssom (POISSON), phân phối T (TDIST) và phân phối Weibull (WEIBULL).

Ngoài các hàm phân phối, tương Excel còn xét các hàm thử định dạng dựa trên các phân phối χ^2 , F, T và Z. Các phép thử này kiểm tra tiêu chuẩn định dạng của các giá trị, hệ số hồi quy, và sự phụ thuộc hay độc lập của các phân phối khác nhau của các giá trị.

1.4.9. Các hàm dữ liệu

Các phép tính trên các hàmi dữ liệu cũng tương tự như các phép tính của các hàm thống kê cơ bản. Tuy nhiên các hìnn dữ liệu áp dụng cho các giá trị được lựa chọn từ cơ sở dữ liệu khi dùng một tiêu chuẩin llựa chọn nào đó. Một cơ sở dữ liệu (database) là một vùng hình chữ nhật trên trang tính Excel. Mỗi dòng trong cơ sở dữ liệu được gọi là một bản ghi (record). Mỗi cột trong; cơ sở dữ liệu được gọi là một trường (field).

Trong bảng 1.8 liệt kê các hànn dữ liệu của Excel, chúng đều có 3 tham số giống nhau bao gồm: tên trương hoặc số hiệu cột của trường chịu tác động của hàm đó, và tiêu chuẩn đề ra. Chỉ những bản ghi niào đáp ứng được tiêu chuẩn đề ra này thì mới được hàm đang gọi xử lý.

Để tiếp cận đến cơ sở dữ liệu trên trang tính, Excel tiếp cận một cơ sở dữ liệu ngoài khi dùng lệnh bộ phát dữ liệu JJET. Các dạng cơ sở dữ liệu phổ biến nhất đều có thể truy vấn được.

| Hàm số | Kết quả tính toán |
|------------|--|
| DAVERAGE() | Số trung bìnhh cong của các giá trị trong trường của các bản ghi thoả mãn tiêu chuẩm |
| DCOUNT() | Số lượng bảản įghi tương cơ sở dữ liệu mà thoả mãn tiêu chuẩn |
| DCOUNTA() | Số lượng; bản t ghi trong cơ sở dữ liệu mà thoả mãn tiêu chuẩn và chứa các giá trị trongg trường, bỏ qua các trường trống rỗng |

Bảng 11.8. Các hàm dữ liệu của Excel

| Hàm số | Kết quả tính toán |
|------------|---|
| DGET() | Khai triển một bản ghi đơn lẻ mà thoả mãn tiêu chuẩn |
| DMAX() | Cực đại của các giá trị trong trường của bản ghi mà thoả mãn tiêu chuẩn |
| DMIN() | Cực tiểu của các giá trị trong trường của bản ghi mà thoả mãn tiêu chuẩn |
| DPRODUCT() | Tích số của các giá trị trong trường của bản ghi mà thoả mãn tiêu chuẩn |
| DSTDEV () | Độ lệch chuẩn mẫu của các giá trị trong trường của bản ghi mà thoả mãn tiêu chuẩn |
| DSTDEVP() | Độ lệch chuẩn phổ thông của các giá trị trong trường của bản ghi mà thoả mãn tiêu chuẩn |
| DSUM () | Tổng của các giá trị trong trường của bản ghi mà thoả mãn tiêu chuẩn |
| DVAR() | Biến sai mẫu của các giá trị trong trường của bản ghi mà thoả mãn tiêu chuẩn |

1.4.10. Các hàm tài chính

Các hàm tài chính rất tiện lợi để theo dõi việc đầu tư hoặc tính toán giá thành, tính toán chi thu tiền bạc, lỗ lãi kinh doanh của một tổ chức hay cá nhân. Bảng 1.9 liệt kê các hàm tài chính của Excel.

| Hàm số | Kết quả tính toán |
|---------------------------|---|
| Các hàm thống kê theo năm | |
| CUMIPMT() | Lợi nhuận tích luỹ giữa thời điểm khởi đầu và kết thúc dự án |
| FV () | Trị số tương lai của một đầu tư hàng năm |
| FVSCHEDULE() | Trị số tương lai của một đầu tư hàng năm sau khi đã xét dãy tỷ suất lợi nhuận |
| IPMT() | Trị số lợi nhuận cho một đầu tư sau một thời kỳ |
| NPER() | Số lượng các chu kỳ đầu tư theo năm |
| PMT() | Tiền trả hàng năm |
| PV () | Giá trị thực của đầu tư hàng năm |
| RATE() | Tỷ suất lợi nhuận thu về hàng năm |

Bảng 1.9. Các hàm tài chính của Excel

Ngoài ra Excel còn có các hàm đầu tư, các hàm thương mại. Bạn đọc là kỹ sư ít dùng đến chúng.

1.4.11. Các hàm thông tin

Chúng ta có thể dùng các hàm thông tin kết hợp với các hàm lôgic để thực hiện nội dung của một ô trong trang bảng tính, để kiểm tra kiểu của ô đó, để tìm vị trí của nó hoặc để tham chiếu. Bảng 1.10 liệt kê các hàm thông tin của Excel.

Hàm TYPE (giá trị) trả về kiểu của giá trị: bằng 1 nếu là con số, bằng 2 nếu là văn bản, bằng 4 nếu là giá trị lôgic, bằng 16 nếu là giá trị sai số, bằng 64 nếu là mảng. Chúng ta có thể dùng TYPE để bảo đảm rằng các giá trị là đúng với kiểu mà ta đã định nghĩa từ trước khi thực hiện phép toám đối với chúng.

Các hàm thử nghiệm lôgic khác như ISBLANK, ISERR, ISERROR và v.v... sẽ thử nghiệm nội dung của giá trị thuộc đúng kiểu đã được định nghĩa từ trước. Chúng trả về giá trị TRUE nếu giá trị là đúng kiểu, kết quả có thể là FALSE nếu không đúng kiểu.

| Hàm số | Kết quả tính toán |
|--------------|--|
| CELL() | Lấy thông tim wề nội dung, định dạng, vị trí của một ô |
| ERRORTYPE () | Số hiệu lỗi, chỉ ra kiểu của lỗi |
| INFO() | Cho thông tim wề môi trường điều hành |
| ISBLANK () | Là TRUE nếu đối số là tham chiếu đến một ô trống |
| ISERR() | Là TRUE nếu đối số là bất cứ lỗi nào, trừ lỗi #N/A |
| ISERROR () | Là TRUE nếu đối số là một trong các lỗi như: |
| | (#N/A,#REF!,#DIV/0!,#NUM], #VALUE!,#NAME?, #NULL!) |
| ISEVEN() | Là TRUE nếu đối số là một số chẩn |
| ISLOGICAL () | Là TRUE nếu đối số là giá trị lôgic |
| ISNA() | Là TRUE nếu đối số là một số hiệu lỗi #N/A |
| ISNONTEXT () | Là TRUE nếu đối số không là chuỗi văn bản |
| ISNUMBER () | Là TRUE nếu đối số là một số |
| ISODD() | Là TRUE nếu đối số là một số lẻ |
| ISREF() | Là TRUE nếu đối số là một tham chiếu ô |
| ISTEXT () | Là TRUE nếu đối số là một chuỗi văn bản |
| N() | Chuyển đổi một giá trị ra con số |
| NA() | Trả về giá trị lõi #N/A |
| TYPE() | Lấy thông tim về kiểu của số đang được lưu trong giá trị |

| Dalig 1.10. Cat ham mong the tua DACC | Bang 1 | .10. | Ciác | hàm | thông | tin | của | Excel |
|---------------------------------------|--------|------|------|-----|-------|-----|-----|-------|
|---------------------------------------|--------|------|------|-----|-------|-----|-----|-------|

1.4.12. Các hàm tham chiếu và hàm tìm kiếm

Các hàm tham chiếu và hàm tìm kiếm cho phép chúng ta quản lý hay xử lý các vùng trên bảng tính. Các hàm này được liệt kê trong bảng 1.11.

1.4.12.1. Các hàm tham chiếu và hàm tìm kiếm véctơ và bảng

Hàm CHOOSE (*chỉ số*, giá trị 1, giá trị 2,...) dùng chỉ số của giá trị để nhặt ra giá trị (hay lựa chọn giá trị) từ một danh sách gồm nhiều giá trị. Nếu chỉ số bằng 1 thì giá trị 1 sẽ được trả về. Nếu chỉ số bằng 2 thì giá trị 2 sẽ được trả về, và v.v... Nếu chỉ số nhỏ hơn 1 hoặc lớn hơn số lượng giá trị thì hàm số sẽ trả về kết quả là #VALUE! Các hàm tìm kiếm là hàm HLOOKUP, VLOOKUP, và LOOOKUP. Người ta dùng chúng để tìm kiếm hàm bảng hoặc bảng dữ liệu đối với các giá trị riêng biệt dịnh sẵn. Nếu tham số *xếp hàm tìm kiếm* bằng TRUE hoặc FALSE thì HLOOKUP (giá trị tìm kiếm, mảng, chỉ số, xếp hàm tìm kiếm) và VLOOKUP (giá trị tìm kiếm, mảng, chỉ số, xếp hàm tìm kiếm) sẽ tìm dòng đầu tiên hoặc cột đầu tiên của *mảng* mà có giá trị lớn nhất rnhưng vẫn còn nhỏ hơn hoặc bằng tham số *giá trị*, rồi chúng chuyển xuống cột hay dòng mà có *chỉ số của ô* rồi trả về giá trị tại đó.

Giá trị ở đây có thể là một giá trị bằng con số, bằng chuỗi văn bản, hoặc giá trị lôgic. Hàm HLOOKUP sẽ tìm kiếm từ trái sang phải sao cho các giá trị trong dòng đầu tiên của mảng được xếp theo trật tự đi xuống. Hàm VLOOKUP sẽ tìm kiếm từ trên xuống dưới sao cho các giá trị trong cột đầu tiên của mảng được xếp theo trật tự đi xuống. Trật tự đi xuống là giá trị số đứng đầu tiên, rồi đến văn bản, rồi đến các giá trị lôgic FALSE, rồi các giá trị lôgic TRUE. Nếu xếp hàm tìm kiếm là FALSE thì các giá trị có thể là theo bất cứ trật tự nào và các hàm sẽ chỉ trả về giá trị nếu chúng tìm thấy một giá trị đúng, nếu không thì sẽ trả về #N/A.

Nếu xếp hàm tìm kiếm là TRUE hoặc không có thì hàm LOOKUP (giá trị tìm kiếm, véctơ 1, véctơ 2, xếp hàm tìm kiếm) sẽ quét véctơ 1 để tìm trị số lớn nhất mà nó còn nhỏ hơn hoặc bằng giá trị và trả về trị số tương ứng từ véctơ 2. Bởi vì với các hàm VLOOKUP và HLOOKUP thì các giá trị trong véctơ 1 cần phải xắp sếp theo trật tự đi xuống, trừ khi mà xếp hàm tìm kiếm là FALSE cho nên các trị số có thể ở trong bất cứ trật tự nào và LOOKUP chỉ trả về giá trị nếu nó tìm thấy phép toán đúng (nếu không thì sẽ trả về #N/A).

Hàm MATCH (giá trị tìm kiếm, véctơ, kiểu) có chức năng tương tự như hàm LOOKUP, chỉ khác là nó trả về chỉ số của giá trị tìm kiếm. Tham số $\langle kiểu \rangle$ xác định bao nhiêu giá trị được so sánh với các giá trị trong véctơ. Nếu $\langle kiểu \rangle$ là 1 thì giá trị nào lớn nhất mà còn nhỏ hơn hoặc bằng giá trị tìm kiếm sẽ được định vị. Nếu $\langle kiểu \rangle$ là -1 thì giá trị nào bé nhất mà còn lớn hơn hoặc bằng giá trị tìm kiếm sẽ được định vị. Nếu $\langle kiểu \rangle = 0$ để tìm các chuỗi văn bản trong chuỗi lộn xộn. Việc định vị một chuỗi có chứa đoạn văn bản đặc biệt sẽ dùng các ký tự x và ?. Hàm sẽ chỉ so sánh phần đầu tiên của chuỗi hoặc đoạn văn bản chứ không xem xét các đoạn sau.

Một trong những hàm tìm kiếm hiệu quả nhất là hàm INDEX (*tham chiếu*, *dòng*, *cột*), nó sẽ trả về giá trị ở ô giao của *dòng* và *cột* của *tham chiếu*. Chúng ta dùng nó để chọn các giá trị trong một mảng tuỳ theo chỉ số của *dòng* và của *cột*. Nếu *tham chiếu* chứa nhiều vùng thì phải thêm một tham số thứ tư nữa vào câu lệnh gọi hàm là *vùng*, để xác định vùng của *tham chiếu*. Nếu tham chiếu là một véctơ đơn thì chỉ cần một chỉ số. Nếu số dòng và số cột là âm thì hàm trả về #VALUE !. Nếu số dòng và số cột yêu cầu là quá lớn thì hàm sẽ trả về #N/A.

1.4.12.2. Các hàm đặc trưng tham chiếu

Hàm AREAS (*ref*) trả về số vùng trong tham chiếu ô *ref*. Một vùng là một khu vực hình chữ nhật liên tục trên bảng tính. Các hàm COLUMN (*ref*), COLUMN (*mảng*), ROW (*ref*) và ROWS (*mảng*) trả về các thông tin về các dòng và các cột trong tham chiếu ô. Hàm COLUMN trả về véctơ chứa các tham chiếu cột đối với mỗi cột trong *ref*. Hàm ROW trả về véctơ chứa các tham chiếu dòng cho mỗi dòng trong *ref*. Hàm COLUMNS và ROWS trả về số cột hoặc số dòng trong *mảng*.

1.4.13. Các hàm bổ sung và các công cụ

Các kỹ sư và các nhà khoa học cần chú ý đến *Analysis Toolpack*, trong đó chứa mọi hàm kỹ thuật bổ sung thêm. Sau đây liệt kê một số hàm đó và một số công cụ thống kê bổ sung cho Excel.

| Chương trình | Mô tả |
|-----------------------------------|--|
| Anova: Hệ số đơn | Phân tích hệ số đơn của biến sai. |
| Anova: Hai hệ số có lặp lại | Phân tích hai hệ số của biến sai có lặp lại |
| Anova: Hai hệ số không lặp lại | Phân tích hai hệ số của biến sai không lặp lại |
| Hiệp tương quan | Tính các hệ số hiệp tương quan |
| Hiệp phương sai | Tính các hiệp phương sai |
| Các thống kê mô tả | Nhiều thống kê về một mẫu |
| Làm mịn hàm mũ | Dự báo với phân tích sai số |
| Thử nghiệm F hai mẫu cho biến sai | Thử nghiệm F hai mẫu |
| Phân tích Fourier | Phân tích truyền Fourier |
| Biểu đồ Histogram | Tính toán dữ liệu cho biểu đồ Histogram |
| Trung bình biến đổi | Trung bình biến đổi của mẫu |
| Bộ phát số ngẫu nhiên | Phát ra các số ngẫu nhiên theo phân phối đã chọn |
| Cấp và tính phần trăm | Bảng xếp hạng của bộ dữ liệu |
| Hồi quy | Hồi quy tuyến tính bội |
| Lấy mẫu | Mẫu của bộ dữ liệu |

Bảng 1.11. Các chương trình công cụ phân tích bổ sung cho Excel

- Thử nghiệm T với hai mẫu thử cho trung bình.

- Thử nghiệm T-Student với hai mẫu thử cho trung bình.

- Thử nghiệm T với hai mẫu thử, giả thiết biến sai bằng nhau.

- Thử nghiệm Z với hai mẫu thử cho trung bình.

1.4.14. Các hàm ngoại

Bên cạnh những hàm nội và các hàm đã được mã hóa trong trang tính Macro, Excel có thể gọi các chương trình con đã lưu trữ trong thư viện liên kết động (DLLs). Nó cũng có thể tiếp gửi đi hoặc nhận các giá trị của các chương trình khác bằng cách dùng trao đổi dữ liệu động (DDE), hoặc dùng Visual Basic for Application và nhúng kết đối tượng (OLE). Khi dùng các công cụ này, bạn có thể tạo ra hàm người dùng trong chương trình khác và rồi gọi chúng từ bên trong Excel.

Để dùng hàm ngoại, hãy khai báo nó trong modul Visual Basic for Application, hoặc dùng hàm CALL (*tệp, hàm, kiểu, đối số*). *Tệp* là tên của tệp DLL, *hàm* là tên của hàm trong tệp DLL, *kiểu* là kiểu dữ liệu của giá trị được trả về, và đối số là các đối số mà hàm sẽ liên quan đến.

Bạn cần phải biết hàm ngoại hoạt động như thế nào để còn gọi nó và dùng nó. Nói chung, cần phải thử nghiệm các giá trị mà bạn định xét để bảo đảm rằng chúng có giá trị và đúng kiểu. Nếu bạn dùng sai đối số, sẽ có thể bị treo chương trình thư viện ngoài và máy không hoạt động. Ưu điểm của các hàm ngoại là chúng chạy nhanh hơn các hàm được VBA điều khiển.

Chương 2

LẬP BẢNG TÍNH TOÁN KHOA HỌC KỸ THUẬT TRÊN EXCEL

2.1. THIẾT KẾ MỘT ỨNG DỤNG KỸ THUẬT - KHOA HỌC TRÊN EXCEL

Cuốn sách này được viết với giả thiết rằng bạn đọc đã có những hiểu biết ban đầu đơn giản nhất về Excel. Vì vậy, các phần dưới đây chỉ tập trung trình bày về việc giải các bài toán kỹ thuật và khoa học. Sẽ không nói lại về những thao tác như mở một bảng tính mới, sao chép và đóng bảng tính hiện hành, v.v...

Để giải một bài toán cần phải tính toán theo một số công thức đã biết, tất nhiên có thể dùng bất cứ ngôn ngữ lập trình nào để giải (Basic, C, Pascal chẳng hạn), nhưng đến khi trình bày kết quả tính toán thì sẽ phải lập trình để tạo ra các bảng sao cho in ra được đẹp mắt và rõ ràng, dễ hiểu. Điều này không dễ dàng và rất tốn thời gian thử đi thử lại. Tuy nhiên nếu bạn dùng Excel để tính toán thì việc in ra kết quả, phân bố nội dung văn bản và hình vẽ, biểu đồ trên cùng một trang in sẽ rất đẹp và dễ dàng. Hơn nữa nếu cần phải tính rất nhiều lần với nhiều giá trị khác nhau của mỗi đại lượng số liệu nhập nhưng cùng dùng chung một số công thức nào đó thì việc lập bảng tính trên Excel là rõ ràng và khoa học nhất.

Chúng ta nên dùng Excel để giải các bài toán khoa học kỹ thuật của mình, chỉ khi nào không thể tính trên Excel được nữa thì mới nên nghĩ đến các công cụ lập trình khác như Pascal, C hay Visual Basic.

Sau đây sẽ trình bày về các bước trong quá trình thiết kế một bảng tính Excel.

2.1.1. Chuẩn bị nội dung kỹ thuật - khoa học của bài toán

Trước khi viết bất cứ chương trình nào và bằng bất cứ ngôn ngữ lập trình nào, người viết cần phải chuẩn bị thật kỹ nội dung kỹ thuật của vấn đề mà mình muốn giải trên máy tính. Không nên vội vàng bắt tay ngay vào việc lập trình khi chưa chuẩn bị kỹ vì như vậy chỉ tốn công sửa chữa và hoàn thiện về sau. Những nội dung cơ bản cần phải chuẩn bị trước khi bắt tay vào viết một ứng dụng kỹ thuật trên Excel như sau:

- Xác định mục tiêu của bài toán, lập bảng danh mục các tham số thể hiện kết quả tính toán (tên gọi, ý nghĩa vật lý, đơn vị đo, trị số ví dụ).

- Lập danh mục các biểu đồ, hình vẽ sơ hoạ thể hiện kết quả (tên gọi, ý nghĩa vật lý, phác thảo dạng).

- Tìm thuật toán giải rồi lập sơ đồ khối thể hiện quá trình tính toán từ khi nhập số liệu đến lúc tính ra kết quả.

- Viết các công thức tính toán theo đúng trình tự tính toán, ghi chú những chỗ có tính toán kiểu chu kỳ lặp.

 Lập bảng danh mục các đại lượng sẽ được nhập vào để bắt đầu tính toán (tên gọi, đơn vị đo, ý nghĩa vật lý, trị số ví dụ).

2.1.2. Thiết kế bảng tính gồm nhiều trang tính

Đối với những bài toán nhỏ thường gặp trong thực tế thì có thể chỉ cần lập bảng tính với một trang tính duy nhất (mặc dù có thể khi in ra sẽ là nhiều trang giấy in). Cách làm này đơn giản, dễ quản lý các công thức và các tham chiếu ô. Tuy nhiên nếu các bảng có chiều rộng cột khác nhau thì rất khó ghép chúng vào chung một trang tính. Trong trường hợp đó thì bạn nên dùng kiểu bảng tính có nhiều trang tính.

Mỗi bảng tính (Workbook) của Excel có nhiều trang tính (Sheet) nhưng thông thường ít khi phải dùng đến quá 6 trang tính (ví dụ không kể các trang tính dùng lưu trữ cơ sở dữ liệu để tra bảng) vì nếu nhiều hơn nữa chỉ thêm khó quản lý và dễ nhằm lẫn lúc viết các công thức có tham chiếu nội dung giữa các trang tính khác nhau. Trên dòng cuối cùng của một bảng tính Excel mới còn trống bạn sẽ thấy các tên *Sheet 1*, *Sheet 2*, *Sheet 3*, v.v... Đó là những tên mặc định của các trang tính trong một bảng tính. Nói chung bạn nên thiết kế bảng tính của mình sao cho gồm các trang tính lần lượt có các nội dung như sau :

- Sheet 1: (sẽ được đổi tên là GioiThieu chẳng hạn) chứa các thông tin ban đầu mà bạn nuốn giới thiệu về tên gọi, công dụng, nội dung và cách sử dụng bảng tính này. Nếu muốt sau này chuyển giao cho người khác cùng sử dụng thì bạn còn nên cho biết tên tác giả, cịa chỉ liên hệ, số điện thoại liên hệ chẳng hạn.

- Sheet 2: (sẽ được đổi tên là NhapSoLieu chẳng hạn) chứa các bảng để nhập các số liệu tan đầu.

- *Sheet 3*: (sẽ được đổi tên là *TinhToan1* chẳng hạn) chứa các bảng để tính toán phần nội dựng kỹ thuật thứ 1.

- *Sheet 4:* (sẽ được đổi tên là *TinhToan2* chẳng hạn) chứa các bảng để tính toán phần nội dung kỹ thuật thứ 2.

Co thể có vài trang tính nữa với nhiệm vụ tính toán tuỳ theo nội dung cụ thể bài toán của tạn (ví dụ Sheet 5, Sheet 6).

- *Sheet 7*: (sẽ được đổi tên là *KetQua* chẳng hạn) chứa các bảng để tính toán phần nội dung kết quả.

Cich đổi tên một trang tính như sau: Giả sử bạn đang ở trang tính Sheet 2, bạn muốn đổi tên thành NhapSoLieu. Hãy nhấn chuột vào mục Format, một thực đơn kéo xuống sẽ
hiện ra, bạn kéo vệt sáng xuống đến dòng có chữ Sheet trên thực đơn đó. Một thực đơn con nữa sẽ hiện ra ở bên phải, bạn nhấn vào dòng Rename. Khi đó tên Sheet 2 ở dòng cuối cùng của màn hình sẽ được tôi đậm, bạn gõ tên mới NhapSoLieu rồi bấm phím Enter là xong.

Tên của các trang tính nên được đặt dựa theo nội dung của trang tính đó để cho dễ nhớ, tất nhiên nên dùng kiểu tiếng V'iệt không dấu và viết liền các chữ cái, riêng chữ cái đầu từ thì nên viết hoa cho dễ nhận ra. Ví dụ như: "NhapSoLieu", "KetQua", v.v...

Nếu ương bài toán của bạn cần phải tra bảng nào đó như bảng các tính năng của dây điện, hay bảng quy cách cốt thép tròm, v.v..., thì nên đặt chúng trong một trang tính riêng và đặt tên là *BangTral*. *BangTra2* chẳng hạn. hoặc cụ thể hơn là *BangDayDien*, *BangCotI hep*, như vậy bạn đã tách riêng các cơ sở dữ liệu để dễ dàng xử lý hơn (sửa đổi, thêm bốt dữ liệu).

2.1.3. Bố trí dạng trang in

Khác với kiểu lập trình trên Pascal hay C. Khi làm việc trên Excel, bạn nên dự kiến định dạng trang in ngay từ đầu. Mỗi trang tính có nội dung riêng nên cách trình bày trang in có thể khác nhau để hình dạng các bảng trong trang tính đó phù hợp nhất với nội dung riêng của nó. Tuy nhiên lúc đầu bạn hãy cứ nên định dạng tất cả các trang tính đều giống nhau Cách làm như sau:

- Giả sử trang tính hiện hành của bạn có tên gọi là *NhapSoLieu*, hãy nhấn chuột vào mục *File* rồi chọn *Page Setup*. Khi đó sẽ hiện ra một khung cửa sổ con như hình vẽ 2.1. Trong khung này phiếu *Page* đang được kích hoạt. Để chọn cách in theo trang đặt theo chiều đứng, bạn hãy nhấn chuột vào ô tròn bên cạnh chữ *Portrait* (nghĩa là kiểu in như in ảnh chân dung). Nếu bạn muốn chọn cách in theo khổ giấy nằm ngang, hãy nhấn vào ô tròn bên cạnh chữ *Landscap* (nghĩa là kiểu in như tranh phong cảnh đặt nằm ngang). Khổ giấy thông dụng ở các tính phía Bắc thường là khổ *A4 (210 × 297 mm)* còn ở các tỉnh phía Bắc thường là khổ *A4 (210 × 297 mm)* còn ở các tỉnh phía Nam thường dùng khổ giấy có tên là "*Letter 8 1/2 × 11 in*". Như vậy bạn sẽ phải nhấn chuột vào ô vuông có ký hiệu mũi tên trong hộp chọn *Paper Size* để tìm đúng khổ giấy mình muốn in.

Đến đây coi như kết thúc việc điều chỉnh trong phiếu Page.

- Tiếp theo bạn hãy nhấn vào phiếu có tên *Margins*, khung cửa sổ con lúc đó sẽ như hình 2.2.. Có 6 ô chọn mà bạn sẽ điều chỉnh để căn lề trái (ô *Left*), căn lề phải (ô *Right*), căn lề đầu trang (ô *Top*), căn lề đáy trang in (ô *Bottom*), định chiều cao tiêu đề đầu trang (ô *Header*), định chiều cao tiêu đề cuối trang (ô *Footer*). Để điều chỉnh các kích thước căn lề bạn chỉ cần nhấn vao các ô mũi tên hướng lên trên (để tăng kích thước) hoặc nhấn vào các ô mũi tên hướng xuống dưới (để giảm kích thước).

| Page Setup | ?× |
|--|---------------|
| Page Margins Header/Footer Sheet | Ĩ |
| Orientation | <u>P</u> rint |
| A Portrait A C Landscape | Print Preview |
| Scaling | Options |
| ፍ Adjust to: 100 🚔 % normal size | |
| C Eit to: 1 💼 page(s) wide by 1 🛖 tall | |
| Paper size: A4 | |
| Print guality: 600 dpi | |
| First page number: Auto | |
| ОК | Cancel |

Hình 2.1: Cửa sổ định dạng trang giấy in khi phiếu Page đang hoạt động.

| Page Setup | | | | ?× |
|----------------|-----------------------|---------|-----------------|---------------|
| Page Margi | ns Header/Footer | Sheet | | |
| | | Header: | |] |
| | | | | Print Preview |
| | | | | Options |
| Left: | | Right: | | |
| | Bottom: | Footer: | | |
| Center on page | | | and shares they | |
| T Horizontal | ly <u>V</u> ertically | | | |
| | | | ОК | Cancel |

Hình 2.2: Cửa sổ định dạng trang giấy in khi phiếu Margins đang hoạt động.

Bây giờ đến mục ghi chú vào đầu trang hoặc cuối trang. Hãy nhấn vào phiếu có tên *Header/Footer*, khung cửa sổ con sẽ hiện ra như hình 2.3. Để soạn nội dung trong mục *Header* bạn hãy nhấn vào núm *Custom Header*, còn để soạn nội dung trong mục *Footer* bạn hãy nhấn vào núm *Custom Footer*. Cách soạn nội dung trong hai mục này tương tự như nhau. Giả sử bạn muốn soạn nội dung mục *Header* ở đầu trang. Sau khi bấm núm *Custom Header*, khung cửa sổ với 3 ô trống sẽ hiện ra như hình 2.4.

| Page Setup | ?× |
|----------------------------------|-----------------------|
| Page Margins Header/Footer Sheet | |
| | Print |
| | Print Previe <u>w</u> |
| Header: (none) | Options |
| Custom Header Custom Footer | |
| Footer: | |
| | |
| 5. C | |
| | |
| 0 | Cancel |

Hinh 2.3: Cửa sổ định dạng trang giấy in khi phiếu Header / Footer dang hoạt động.

| Header | | | ?X |
|---|-----------------|---------------|--------|
| To format text: select the text, then To insert a page number, date, time, insertion point in the edit box, the To insert picture: press the Insert Pic cursor in the edit box and press th | OK Cancel | | |
| A left section: | Center section: | Sight section | |
| | | | * - |
| | 9 | 2 | |

Hình 2.4: Cửa sổ soạn nội dung in ở đầu trang và cuối trang.

Các nội dung nên điền vào các phần này thường là:

- Tên tệp của bảng tính, bạn hãy gõ ký hiệu &[File] hoặc nhấn vào biểu tượng của tên tệp.

- Số hiệu của trang in, bạn hãy gõ ký hiệu &[Page], hoặc nhấn vào biểu tượng của số hiệu của trang in.

- Số lượng các trang in của trang tính mà đang xét, bạn hãy gõ ký hiệu & [Pages], hoặc nhấn vào biểu tượng của số lượng các trang in

- Ngày và thời gian in ra giấy, bạn hãy gõ &[Date], hoặc nhấn vào biểu tượng của ngày và giờ in các trang in.

Tất nhiên vì trong khung cửa sổ có tới 3 ô trống để soạn thảo nội dung cần in vào phía bên trái, hoặc in vào chính giữa hoặc in vào phía bên phải của trang in, nên bạn muốn gõ các ký hiệu nói trên vào cùng 1 ô hay vào 3 ô riêng rẽ cũng được tuỳ ý.

Sau khi soạn thảo xong, hãy nhớ đóng từng cửa sổ bằng cách nhấn vào phím bấm OK trên cửa sổ đó. Đến đây coi như đã kết thúc việc chuẩn bị định dạng hình thức của các trang in. Công việc này bạn phải làm cụ thể lần lượt cho từng trang tính trong bảng tính của bạn.

2.1.4. Soạn thảo nội dung trang tính dành để nhập số liệu

Sau khi đã chuẩn bị xong nội dung danh mục các số liệu cần nhập trên giấy nháp, kích hoạt trang tính *NhapSoLieu* và bắt đầu biên soạn các bảng trong trang tính đó để nhập số liệu. Nói chung, mỗi loại số liệu nên nhập vào một bảng riêng, ví dụ như:

- Bảng nhập các số liệu hình học (chiều cao, chiều rộng, v.v...).

- Bảng nhập các đặc trưng vật lý (độ dẫn nhiệt, môđun đàn hồi, v.v...).

- Bảng nhập các đặc trưng cơ học (cường độ chịu lực, độ dãn dài tới hạn,v.v...).

- Bảng nhập các trị số giới hạn cho phép (độ võng, nhiệt độ, tần số dao động, độ rộng vết nứt, hàm lượng tạp chất độc hại cho phép, v.v...).

Đến đây có thể xuất hiện một vấn đề là kích thước hợp lý nhất của các ô trong các bảng số liệu nhập có thể không giống nhau. Như vậy phải thoả hiệp, chọn các kích cỡ bề rộng các ô trong bảng tính Excel sao cho phù hợp được với mọi bảng số liệu cần thể hiện. Gặp trường hợp khó xử lý quá thì có thể lập thêm 1 trang tính nữa dành cho việc nhập số liệu với tên gọi *NhapSoLieu2* chẳng hạn và cố định bề rộng cột khác với ở trang tính *NhapSoLieu1* sao cho phù hợp nhất với nội dung của bảng các số liệu nhập nào đó đang xét.

Để đảm bảo trang in gọn cả bề rộng lẫn bề dài thì trước lúc soạn thảo bảng số liệu nhập, hãy gõ tên của một bảng số liệu nhập vào một ô nào đó của trang tính tuỳ ý bạn chọn, sau đó hãy chọn mục *Print Preview* trên thanh thực đơn của bảng tính Excel. Khi

đó trên trang tính sẽ xuất hiện các đường kẻ nét chấm chấm thẳng đứng và đường kẻ nét chấm chấm nằm ngang báo cho bạn biết mép của trang in để bạn cân nhắc bố trí hình thức các bảng cho phù hợp gọn trong khuôn khổ đã định.

Nếu bảng số liệu của bạn quá rộng khiến cho chúng nhô ngang ra khỏi đường kẻ nét chấm chấm thẳng đứng nói trên thì bạn có thể thu nhỏ bề rộng một số cột nào đó. Khi đó, cỡ chữ trong các ô cũng có thể thu nhỏ, ví dụ như từ cỡ chữ 12 xuống cỡ 8 để vẫn xem được. Nếu vẫn không xong thì giải pháp cuối cùng sẽ là phải định dạng cho trang in theo khổ giấy nằm ngang, tức là nhấn vào ô của mục *Landscap* trong phiếu *Page* của *File/Page Setup*.

Nếu bảng số liệu của bạn quá dài khiến cho chúng nhô xuống quá khỏi đường kẻ nét chấm chấm nằm ngang nói trên, thì cách tốt nhất là chuyển bảng số liệu đó sang trang in tiếp theo, mặc dù sẽ phải để một số dòng trống trong trang tính (Sheet). Tuy nhiên bạn có thể viết các lời giải thích gì đó hay ghi công thức vào các dòng trống nói trên cho khỏi phí giấy in.

Một giải pháp khác là thu nhỏ độ cao của mỗi dòng của trang tính và cỡ chữ trong chúng để cùng trên một trang in có thể in được nhiều dòng hơn. Cách làm như sau:

- Trước hết hãy thu nhỏ kích thước hiển thị trang tính trên màn hình, giả sử bạn đang dùng mức độ hiển thị 100%, hãy nhấn chuột vào ô phần trăm đó ở bên phải của thanh thực đơn ở trên cùng của màn hình Excel, chọn mức hiển thị 75% hoặc 50% sao cho vừa mắt để bạn có thể sắp xếp các bảng mà không cần xem nội dung chi tiết của từng ô trong bảng. Khi đó các đường biên nét chấm chấm của mỗi trang in sẽ hiện ra rất rõ.

- Nhấn chuột vào ô trên cùng bên phải của trang tính (bên trái cột A, bên trên dòng số một), khi đó toàn bộ trang tính sễ được chọn và bối mầu xám. Di con trỏ chuột xuống đến biên giữa 2 dòng bất kỳ trên cột biên trái của màn hình, con trỏ sẽ biến thành một dấu điều chỉnh độ cao dòng, hãy rê chuột lên trên cho thu nhỏ độ cao dòng lại, tất cả các dòng khác của trang tính sẽ thu nhỏ theo giống nhau. Bạn sẽ làm thử vài lần cho đến khi hài lòng vì bố trí được bảng gọn trong trang in.

- Nếu không muốn rê chuột để điều chỉnh độ cao cột, bạn cũng có thể dùng bàn phím để đạt mục đích. Hãy bấm vào núm Format trên thanh thực đơn của màn hình Excel. Khi đó hiện ra một thực đơn kéo xuống, hãy chọn Row, lại một thực đơn nữa hiện ra bên phải, hãy bấm vào ô Height cho hiện ra khung cửa sổ điều chỉnh độ cao cột, rồi gõ trị số độ cao dòng mà bạn muốn vào ô bên phải chữ Row Height, sau đó nhấn chuột vào núm OK là xong.

Trên hình 2.5 là vài dòng của một mẫu biểu đơn giản để nhập số liệu của tính toán độ lún móng đê đắp trên nền đất yếu.

| STT | Tên số liệu nhập | Ký hiệu | Đơn vị đo | Giá trị nhập | Giá trị ví dụ |
|-----|----------------------------|----------------|--------------------|--------------|---------------|
| 1 | Chiều cao đắp đê | Н | m | 6 | 1-10 |
| 2 | Góc ma sát trong của nền | φ | độ | 23 | 5-35 |
| 3 | Lực dính của đất nền | C | kg/cm ² | 0,49 | 0,5 |
| 4 | Chỉ số nén lún của đất nền | C _c | cm²/kg | 0,22 | 0,3 |
| | | •• | | ••• | |

Hình 2.5: Ví dụ bố trí nội dung trang tính nhập số liệu (tính lún đê).

Trên một bảng số liệu nhập bất kỳ nên có đủ các cột sau:

- Số thứ tự.
- Tên gọi và ý nghĩa của số liệu được nhập.
- Ký hiệu toán học hoặc ký hiệu trong chương trình của số liệu.
- Đơn vị đo.

- Giá trị bằng số sẽ được nhập (ô này nên tô màu xám nhạt để khi in dễ phát hiện, hoặc ghi chữ bằng mầu xanh nước biển, hoặc kẻ khung viền bằng nét đậm để dễ nhận ra trên màn hình).

- Giá trị ví dụ bằng số để người sử dụng dễ phát hiện sai lầm, tránh nhập con số vô lý. Ô này nên dùng mầu chữ nhạt (xanh lá mạ chẳng hạn) để chỉ thấy mờ mờ trên màn hình.

Ghi chú: Trong trang tính dùng để nhập số liệu còn nên có các sơ đồ hoặc hình vẽ để giải thích về ý nghĩa của các số liệu nhập đó. Để tạo ra các hình vẽ trên Excel có thể dùng các công cụ về của chính Excel hoặc vẽ trên AutoCAD hay Paint của Microsoft Office rồi dùng kỹ thuật nhưng kết đối tượng mà chèn vào trang tính của Excel. Một chương của cuốn sách này sẽ trình bày cách vẽ hình và chèn vào bảng tính. Một chương riêng khác sẽ giới thiệu cách tạo ra các biểu đồ trên bảng tính.

2.1.5. Soạn thảo nội dung trang tính dành để xuất kết quả

Về cơ bản, cách soạn thảo nội dung trang tính dành để xuất kết quả cũng giống như khi soạn thảo trang tính dành để nhập số liệu. Các kết quả tính toán cũng cần được phân loại trước và xếp vào từng bảng theo nhóm ý nghĩa để dễ dàng theo dõi và kết luận. Cấu tạo một bảng kết quả cũng nên gồm các cột có nội dung gần giống bảng mẫu về số liệu nhập (hình 2.5). Để nhấn mạnh và dễ nhận thấy nên cho hiện các kết quả bằng chữ mầu đỏ.

Kèm theo các bảng ghi kết quả nên có các hình vẽ minh hoạ. Nếu có thể nên cố gắng vẽ các biểu đồ minh hoạ kết quả tính toán theo số liệu thực. Như vậy khi kết quả thay đổi thì các biểu đồ và hình vẽ cũng được tự động thay đổi theo tương ứng. Nếu điều này không thực hiện được thì đành vẽ bằng công cụ của Excel, của Paint hay AutoCAD nhưng khi đó chỉ thuần tuý là hình vẽ minh họa mà không mô tả được kết quả thực.

Hầu như mọi kết quả đã được tính ra từ các trang tính dành riêng cho tính toán, vì vậy nảy sinh vấn đề tham chiếu các ô của các trang tính trước đó. Cách viết công thức có tham chiếu đến các trang tính khác hoặc bảng tính khác đã được trình bày trong chương 1.

2.1.6. Soạn thảo nội dung trang tính dành để lưu trữ cơ sở dữ liệu

Hầu hết các tính toán khoa học kỹ thuật đều có dùng đến các bảng tra, các đồ thị khác nhau đã được soạn sẩn. Như thế chúng ta phải lưu trữ các bảng tra này trong những trang tính riêng coi như cơ sở dữ liệu để tra cứu rồi nội suy hoặc ngoại suy phục vụ tính toán. Ví dụ để tính toán kết cấu thép, cần phải có các bảng tra quy cách các loại thép hình đã sản xuất sẵn. Để tính toán giá thành của một sản phẩm nào đó sẽ cần đến bảng đơn giá các nguyên vật liệu và giá gia công, v.v...

Trước hết hãy đổi tên một trang tính (ví dụ từ *Sheet-4* thành *DuLieu1* chẳng hạn) rồi sao chép y nguyên nội dung bảng tra mà bạn vẫn dùng khi tính toán bằng tay vào trang tính đó. Bây giờ vì bạn không cần in ra bảng tra cho nên không cần quan tâm đến định dạng in cho trang tính này, nó có bề rộng và bề dài tuỳ theo thực tế nội dung của nó. Tuy nhiên nên nhớ thêm một cột đầu tiên bên trái cho số thứ tự của các dòng trong mỗi bảng cơ sở dữ liệu này. Như thế sẽ tiện theo dõi hơn.

2.1.7. Soạn thảo nội dung các trang tính dành để tính toán

Câu hỏi đầu tiên là bạn cần bao nhiêu trang tính dành cho tính toán thì hợp lý nhất? Thật khó trả lời. Nếu có thể thì chỉ cần 1 đến 2 trang tính (Sheet) dành cho tính toán. Nếu hơn nữa càng thêm khó quản lý mà thôi. Tuy nhiên nếu bài toán quá lớn thì cũng có thể dành mỗi trang tính cho mỗi nội dung tính toán riêng để tiện theo dõi cũng như dễ bố trí trang giấy in ra cho đẹp.

Trong các chương sau sẽ có những ví dụ về chương trình tính toán cho một số bài toán kỹ thuật điển hình. Điều đáng lưu ý là nên dành chỗ cho các lời giải thích và ghi các công thức đã được dùng trong các bảng tính toán. Vị trí này nên ở ngay bên trên hay bên dưới của bảng tính toán có liên quan để người xem dễ theo dõi. Điều phức tạp ở đây là Excel không thể hiện được các công thức toán học theo kiểu như sách in, việc này thì Microsoft Word thực hiện tốt hơn.

Như vậy bạn có thể soạn thảo các công thức trong Word rồi nhúng vào trang tính của Excel coi như một ảnh đồ hoạ (bằng cách sử dụng lần lượt các lệnh *Copy* của Word và *Paste Special* của Excel).

Việc bố trí hợp lý các bảng trong trang tính nói chung khó mà đặt ra ngay từ đầu vì bạn còn đang trong quá trình suy nghĩ dần dần và phát triển dần bảng tính của mình. Hãy cứ soạn các bảng theo nội dung kỹ thuật cần thiết cho đến khi xong việc hoàn toàn nghĩa là tính ra đến kết quả cần thiết cuối cùng. Để viết các công thức trong các ô tính toán bạn chỉ cần dùng các toán tử và các hàm đã được trình bày trong chương 1, kết hợp với cách tham chiếu ô ở các trang tính khác như đã hướng dẫn.

Sau khi đã tính ra đến kết quả cuối cùng, bạn hãy sửa sang các bảng của phần tính toán cho gọn và dễ in ra trong khổ trang in đã dự kiến. Đối với những bảng rộng quá khổ giấy in thì hãy thử bố trí riêng nó trên một trang tính riêng và định dạng cho trang được in ra theo khổ nằm ngang cho đủ rộng (kiểu dạng in *Landscap*). Một cách khác là chuyển bớt một số cột nào đó trong bảng mà không nhất thiết phải in (hoặc chính bạn muốn giấu không in các cột đó ra giấy) ra biên bên phải của bảng đang xét và đóng khung chúng thành một bảng phụ mới nằm ngay bên phải bảng chính nhưng ở bên phải đường nét chấm chấm thể hiện mép bên phải của trang giấy in. Như vậy kết quả tính toán vẫn không thay đổi nhưng bảng tính toán phụ này sẽ thuộc về một trang giấy in khác và có thể được in ra hay không là tuỳ theo ý bạn. Bảng tính toán chính lúc này sẽ đủ gọn phù hợp với khổ rộng của tờ giấy in.

2.2. GIẢI PHƯƠNG TRÌNH ĐƠN GIẢN

2.2.1. Tính theo cóng thức thông thường

Giả sử chúng ta cần phải giải một phương trình bậc 2 có dạng như sau:

 $5x^2 - 6x + 7 = 0$

Dạng của phương trình này là $Ax^2 + Bx + C = 0$

Công thức tính nghiệm x_1 và x_2 của phương trình này đã được giới thiệu trong trường phổ thông và chúng ta có thể gõ các công thức đó vào 2 ô nào đó để tính ra 2 nghiệm. Đây là cách làm thông thường nhất, có thể lập thành bảng để tính với nhiều phương án khác nhau của các giá trị hệ số A, B, C như sau:

| Giá trị của các hệ số A, B, C | | Delta | Nghiệm x ₁ | Nghiệm x ₂ | Ghi chú | |
|-------------------------------|----|-------|-----------------------|-----------------------|---------|--|
| A | В | С | | | | |
| 5 | -6 | 7 | | | | |
| 3 | 12 | 4 | | | | |
| 65 | 34 | - 2 | | | | |

2.2.2. Dùng hàm Goal Seek và hàm Solver

Tuy nhiên có một cách khác để giải phương trình nói trên: sử dụng 2 công cụ rất mạnh của Excel là các hàm GOAL SEEK hoặc SOLVER.

Chức năng của GOAL SEEK là thay đổi giá trị của biến số (nằm trong ô *By changing cell*) và của hàm mục tiêu (nằm trong ô *Target cell*) để đạt tới một giá trị đích (to value)

xác định nào đó. Như vậy, hàm GOAL SEEK sẽ tìm nghiệm của một phương trình f(x) = 0 bất kỳ, trong đó x là biến số, f(x) là hàm mục tiêu và 0 (hay một giá trị nào đó) là giá trị đích.

Sau đây xin giới thiệu lại ví dụ trong bài báo của tác giả Võ Quang Huy đăng trên Tạp chí Thế giới vi tính 1997 hướng dẫn trình tự giải phương trình y = $f(x) = x^2 - 4x + 3 = 0$

- Bước 1: Trước tiên cần phải lập một mô hình bao gồm biến số và hàm mục tiêu trên bảng tính như hình 2.6 sau:

Chọn một ô bất kỳ để chứa biến số, ví dụ ô A2, cho biết giá trị biến x = 0 chẳng hạn.

| | А | В | | C C |
|---|---------|--------|---|-----|
| 1 | Biến số | Hàm số | | |
| 2 | 0 | | 3 | |
| З | | | | |
| 4 | | | | |

Hình 2.6: Lập mô hình hàm số.

Chọn một ô khác để chứa hàm số, ví dụ ô B2, sau đó gõ công thức tính hàm số vào ô này như sau "= $A2^2 - 4^*A2 + 3$ ". Trong đó A2 là thừa chú tham chiếu tới ô chứa biến

(gọi tắt là ô biến) và B2 là thừa chú tham chiếu của hàm mục tiêu (gọi tắt là ô hàm).

- Bước 2: Tiếp theo bạn hãy gọi hàm GOAL SEEK và gán các thông số vào theo yêu cầu. Bấm vào núm *Tools* trên thanh thực đơn, một thực đơn kéo xuống, trên đó hãy chọn GOAL SEEK. Hộp thoại GOAL SEEK xuất hiện như hình 2.7.

Có 2 cách nhập các thông số vào hộp thoại này:

| Goal Seek | ?× |
|-------------------|--------|
| Set œll: | 83 |
| To <u>v</u> alue: | |
| By changing cell: | |
| 10 | Cancel |

Hình ?.7: Hộp thoại Goal Seek.

- Cách thứ nhất là: đưa tham chiếu trực tiếp vào hộp thoại của GOAL SEEK bằng cách nhấn chuột vào các ô trong bảng, bao gồm các thao tác sau đây:

+ Đưa thừa chú của ô hàm số là ô \$B\$2 vào ô Set cell như trong hình 2.7 bằng cách nhấn chuột trong khung Set cell của hộp thoại rồi nhấn chuột và ô B2 của bảng tính.

+ Nhập giá trị của vế phải của hàm (tức là số 0) vào bằng cách đánh trực tiếp số 0 vào khung *To value*.

+ Đưa thừa chú của biến số là \$A\$2 vào bằng cách nhấn chuột trong khung By changing cell của hộp thoại rồi nhấn chuột vào ô A2 như trong hình 2.6.

- Cách thứ hai là: đưa tên của các ô (sau khi đã định nghĩa tên) vào các tham số của GOAL SEEK.

+ Trước tiên cần phải đặt tên cho ô chứa biến số (tức là ô A2) là X và ô chứa hàm số (tức là ô B2) là Y.

+ Gõ tên của các biến số và hàm số (X và Y) trực tiếp vào hộp thoại.

+ Sau khi nhập tên xong bạn có thể nhập vào ô B3 công thức sau = $X^2 - 4*X + 3$, như thế gần giống với công thức thông thường và dễ theo dõi hơn.

+ Tuỳ theo giá trị nhập vào ô A2 mà chúng ta có các giá trị kết quả hàm số khác nhau.

 Bước 3: Bây giờ bạn hãy thực hiện hàm GOAL SEEK bằng cách nhấn OK. Kết quả xuất hiện trong hình 2.8.

GOAL SEEK dựa vào phương pháp tính gần đúng, nên thường cho kết quả. Trong vế dưới bài toán có nghiệm nguyên vào GOAL SEEK cho ra kết quả với nhiều số thập phân.

| | A | В | С |
|---|---------|---------|---|
| 1 | Biến số | Hàm số | |
| 2 | 0.99973 | 0.00053 | |
| З | | | |
| 4 | | | |

Hình 2.8: Kết quả sau khi dùng Goal Seek.

Khi đó nếu làm tròn kết quả, bạn hãy chọn trên thanh thực đơn của Excel theo trình tự sau: *Format / Cells / Numbers / Decimal Preses* rồi bấm vào mũi tên để điều chỉnh cho số lượng chữ số sau dấy phẩy thập phân bằng 0, kết quả như trong hình 2.9.

| | ivnArial | | 12 - B | u ∣≣= |
|---|----------|-------------------------|---------------|-------|
| | A2 | ▼ f _x | 0.9997327183 | 15683 |
| | A | В | C | D |
| 1 | Biến số | Hàm số | | |
| 2 | 1.0 | 0.00053 | | |
| 3 | | Ť | | |
| 4 | | | | |

Hình 2.9: Kết quả sau khi làm tròn số.

Như vậy, bạn tìm được một nghiệm là 1. Tuy nhiên, phương trình bậc hai có hai nghiện, cần tìm nghiệm thứ hai như sau:

+ Ước lượng phạm vi của nghiệm thứ hai này, ví dụ khoảng chia nghiệm là (2,5).

+ Chọn giá trị khai báo cho biến X trong khoảng nghiệm ước lượng, chẳng hạn 4.

Tương tự như vậy ta lại tìm thêm một nghiệm khác là 3 sau khi đã làm tròn.

Như vậy là bằng cách dùng GOAL SEEK, sau hai lần tìm, nghiệm của phương trình bậc hui là 1 và 3. Ngoài ra, có thể sử dụng chức năng SOLVER tham gia vào cho kết quả tương tự. Nhưng phương trình phức tạp hơn các phương trình bậc 3, phương trình siêu việt, 7.v... cũng có thể giải được theo cách tương tự một cách dễ dàng. Tuy nhiên phải chọn giá trị khai báo của biến số là phù hợp.

| | B2 | ✓ f _x = | A2^2-4*A2+3 | | | | |
|----|---------|--------------------|----------------|----------------|----|--------------|-----|
| | A. | В | C | D | E | F | G |
| 1 | Biến số | <u>Hàm số</u> | | | | | |
| 2 | 3.0 | 0 | | | | | |
| 3 | | | Frank Sealt St | - | | 1 | |
| 4 | | | Guar seek si | | | L | |
| 5 | | | Goal Seeking | with Cell B2 | | ОК | |
| 6 | | | may not have f | ound a solutio | n. | (| i i |
| 7 | | 1 | Target value: | Û | | | |
| 8 | | | Current value: | 0 | | Ste p | |
| 9 | | | | | | Danaca | 1 |
| 10 | | | | | | | i |

Hình 2.10: Kết quả tính nghiệm thứ hai của phương trình bậc hai.

2.3. MỘT SỐ THỦ THUẬT BỔ SUNG

2.3.1. Tính giá trị trung bình dãy số khi có những giá trị bằng không

Để tính giá trị trung bình của một dãy số, Excel cung cấp hàm AVERAGE (). Tuy nhiên nếu trong dãy số liệu ghi trong một cột nào đó của bảng tính mà có những giá trị bằng 0 thì việc dùng hàm AVERAGE sẽ cho kết quả sai. Để tránh điều này có thể tuỳ chọn một trong hai cách sau đây:

- Cách thứ nhất: vì hàm AVERAGE bỏ qua không xét đến các ô nào chứa chuỗi các ký tự hay khoảng trắng nên chúng ta sẽ thay giá trị 0 trong các ô nào đó bằng một khoảng trắng. Ví dụ nếu trong ô nào đó đang có công thức = A1 + B2*C3 chẳng hạn, bạn hãy sửa lại thành công thức = IF (A1 + B2*C3 = 0,"", A1 + B2*C3), như vậy nếu kết quả tính toán bằng 0 thì trong ô đó sẽ chứa khoảng trắng, còn nếu kết quả khác 0 thì sẽ hiện ra trong ô đó.

- Cách thứ hai: nếu bạn vẫn muốn trong ô luôn xuất hiện kết quả tính toán, dù là bằng 0 thì nên dùng hàm COUNTIF () với thủ thuật là ra lệnh cho tính tổng các giá trị của các ô trong phạm vi đã chỉ định của cột hoặc dòng rồi chia tổng đó cho số lượng các giá trị khác 0 đã được xét. Ví dụ đối với yêu cầu tính giá trị trung bình của các giá trị trong các ô từ A1 đến A9 rồi ghi kết quả vào ô A10 thì công thức ở ô A10 nhằm tính giá trị trung bình sẽ là:

= SUM (A1:A9)/COUNTIF(A1:A9,"<>0")

2.3.2. Tính toán lại hoặc xem kết quả tính từng phần của công thức

Nếu sau khi viết một công thức bạn muốn xem kết quả của nó, hãy nhấn phím F9. Nếu muốn đặt ngay kết quả tính toán vào ô hiện hành thì bấm phím Enter, nếu muốn quay trở lại công thức thì bấm phím Esc. Hơn nữa nếu công thức rất dài mà bạn muốn tính toán thử chỉ một phần trong công thức đó thì bạn chỉ cần điểm sáng đoạn công thức cần chọn rồi nhấn phím F9.

2.3.3. Có thể thêm ghi chú vào công thức để dễ hiểu

Để dễ theo dõi nội dung bảng tính, chúng ta nên thêm các ghi chú vào các công thức của bảng tính. Bạn chỉ cần chọn thực đơn *Formula / Note* rồi nhập các ghi chú vào.

2.3.4. Công thức cho' biết tên file

Đôi khi chúng ta cần cho tên file xuất hiện trong một ô nào đó của bảng tính. Cách làm như sau:

- Nếu muốn biết cả đường dẫn và thư mục, hãy gõ công thức = info ("directory").

- Nếu chỉ cần biết tên file, hãy gõ công thức:

= mid (cell(" filename"), len(inforectory")) + 1,12).

Chương 3

VẼ ĐỒ THỊ THEO DỮ LIỆU VÀ CÁC HÀM SỐ TRÊN EXCEL

Trong chương này, bạn sẽ học cách tạo ra các biểu đồ của dữ liệu bảng tính và nâng cấp các biểu đồ này với các nhãn dữ liệu, các dấu hiệu, lời chú giải và các tiêu đề. Ngoài các chức năng vẽ đồ thị thông thường này, bạn sẽ tìm ra cách sử dụng các khả năng vẽ đồ thị của Excel để thực hiện các nhiệm vụ vẽ đồ thị khoa học và kỹ thuật thông thường khác, chẳng hạn như, tạo ra các lôgarit hay bán lôgarit, vẽ các đồ thị không gian ba chiều, và nâng cấp các đồ thị này bằng các thao tác vẽ đơn giản.

Sử dụng các kỹ thuật đã trình bày trong chương này một cách linh hoạt, bạn sẽ có thể quản trị được hầu hết các công tác trình bày và vẽ đồ thị dữ liệu. Đối với kiểu đồ thị mới nào đó, chẳng hạn như đồ thị cực, hãy tạo ra một phép biến đổi tuyến tính để thay đổi dữ liệu của bạn lên trên các trục x và y tuyến tính trong Excel. Vẽ đồ thị dữ liệu của bạn, và sau đó sử dụng các công cụ của Excel để bổ sung thêm bất kỳ các đường trục hoặc các nhãn được yêu cầu nào.

Trong công tác hàng ngày, các kỹ sư cũng như các nhà nghiên cứu luôn làm việc với các dữ liệu số. Đó là những kết quả tính toán hoặc kết quả điều tra, đo đạc trong phòng thí nghiệm hoặc tại hiện trường, nhà máy. Sau đó cần phải phân tích các dữ liệu đó để tìm ra những quy luật và những kết luận cần thiết. Hình thức phân tích đơn giản nhất là vẽ đồ thị theo các dữ liệu số đó và phân tích thử xem chúng có dạng đường cong nào, có thể biểu diễn bằng loại phương trình gì. Bằng cách xem xét hình dạng của một đồ thị dữ liệu, người ta đã thu được rất nhiều kết luận khoa học sâu sắc. Trong chương này chúng ta sẽ giới thiệu khả năng vẽ đồ thị của Excel và cách áp dụng chúng sao cho phù hợp với các ứng dụng khoa học và kỹ thuật của mỗi người.

3.1. CÁC KIỂU ĐỒ THỊ TRONG EXCEL

Trong Excel có thể biểu diễn 2 kiểu đồ thị:

- Kiểu đồ thị vẽ ngay trên trang bảng tính bên cạnh bảng ghi các dữ liệu liên quan.

- Kiểu đồ thị vẽ trên những trang đồ thị riêng rẽ với các dữ liệu liên quan.

Các đồ thị vẽ trong trang tính Excel là việc tự nhiên và đơn giản nhất. Còn các trang đồ thị riêng rẽ là những tài liệu Excel riêng chứa một đồ thị đơn lẻ. Tuy nhiên ở cả hai kiểu đồ thị người ta nối kết các điểm được vẽ trên đồ thị với các ô của một bảng tính. Tuỳ theo nội dung công việc cụ thể mà bạn sẽ chọn kiểu đồ thị thích hợp nhất với nó.

Khi đang phân tích dữ liệu trên một trang bảng tính, tất nhiên bạn sẽ thích sử dụng đồ thị nhỏ đã được vẽ ngay bên cạnh bảng số liệu. Các đồ thị nhỏ này thường quá nhỏ đôi với mục đích trình bày in đẹp nhưng chúng lại phù hợp đối với các mục đích phân tích quy luật của số liệu. Nếu chúng ta cần trình diễn đồ thị này trong Hội nghị thì tốt nhất là sử dụng trang riêng cho đồ thị để tạo ra một đồ thị lớn, chất lượng tốt cho in và trình diễn.

3.2. SU DUNG CHART WIZARD

Việc tạo ra một đồ thị trong Excel thường đòi hỏi phải thực hiện bốn hoặc năm lệnh khác nhau để tạo ra đồ thị cơ bản, thiết đặt kiểu của nó, thêm các nhãn và v.v... Để đơn giản hoá quá trình này, Excel-97 cung cấp *Chart Wizard* - một lệnh đơn lẻ đưa bạn đi qua lần lượt từng các bước cần thiết để tạo ra một đồ thị.

Chart Wizard có 5 khung cửa sổ chuẩn hoá, mỗi cửa sổ tập hợp nhiều thông tin cần thiết cho việc tạo đồ thị. Bước đầu tiên trong việc tạo một đồ thị là lựa chọn dữ liệu được vẽ đồ thị, như sau:

- Nếu cột dữ liệu x và một hoặc nhiều cột dữ liệu y ở kề sát nhau thì hãy chọn các cột này chung 1 khối.

 Nếu ô đầu tiên trong mỗi cột chứa đoạn Text sử dụng trong lời chú giải thì cũng chọn luôn những ô này.

- Nếu các cột dữ liệu không ở gần kề nhau, hãy chọn cột dữ liệu x trước. Sau đó, ấn phím Ctrl, và chọn cột dữ liệu y đầu tiên. Tiếp tục chọn các cột y tiếp theo cho đến khi bạn đã chọn đủ dữ liệu cần thể hiện trên đồ thị.

- Nếu dữ liệu x và y ở trong các hàng thay vì trong các cột, thì chọn dữ liệu như đã mô tả ở (*), nhưng theo các hàng thay vì theo các cột.

Sau khi chọn dữ liệu, hãy chọn lệnh Chart Wizard (nút này trông giống như một đồ thị nhỏ) trên thanh công cụ. Khi bạn nhấn vào nút này, cửa sổ đầu tiên của Chart Wizard xuất hiện như thể hiện trong hình 3.2. Cửa sổ đầu tiên này dùng để chọn kiểu đồ thị mà bạn muốn thể hiện.

| 11 | A | В | C | D |
|----|----|----|----|------|
| 1 | X | y1 | y2 | у3 |
| 2 | 1 | 10 | 13 | 20 |
| 3 | 3 | 12 | 17 | -36 |
| 4 | 5 | 15 | 22 | - 38 |
| 5 | 7 | 18 | 24 | 41 |
| 6 | 14 | 20 | 29 | 48 |
| 7 | 28 | 22 | 30 | 51 |
| 8 | 56 | 23 | 32 | 53 |

Hình 3.1: Bước đầu tiên của bạn là chọn dữ liệu cho đồ thị.

3.2.2. Chọn kiểu đồ thị

Trong cửa sổ bên trái của hình 3.2 của Chart Wizard, bạn hãy chọn kiểu đồ thị từ 14 kiểu có sẵn. Trong tất cả các kiểu đồ thị này, chỉ có các đồ thị dạng XY và 3-D Surface là thực sự có ích cho mục đích khoa học và kỹ thuật. Đôi khi còn dùng kiểu LINE.

Kiểu đồ thị XY là một đồ thị vuông góc, với các thang tỷ lệ xích lôgarit hoặc tuyến tính trên các trục. Chúng ta cần lưu ý rằng kiểu đồ thị XY là kiểu duy nhất mà trên thực tế vẽ đồ thị vùng dữ liệu x. Tất cả các kiểu đồ thị khác sử dụng dữ liệu x cho các nhãn, vì vậy cho dù bạn sử dụng bất kỳ giá trị nào cho dữ liệu x, thì tất cả các điểm được vẽ đồ thị đều được đặt cách đều nhau theo phương x. Trừ khi toàn bộ dữ liệu của bạn được đặt cách và được sắp xếp đều nhau, nếu không các kiểu đồ thị khác không đặc biệt hữu ích cho các nhiệm vụ khoa học và kỹ thuật.

Đồ thị mặt 3-D (3 chiều) vẽ một lưới dữ liệu hình chữ nhật dưới dạng mặt 3-D. Dữ liệu cho các trục x và y là ở bên trên và phía phải của lưới dữ liệu z. Đồ thị này chỉ vẽ các giá trị dữ liệu z và đặt các điểm được vẽ cách đều nhau dọc theo phương x và y. Dữ liệu x và y được sử dụng chỉ cho các nhãn. Nhấn chuột trên kiểu đồ thị mà bạn muốn tạo ra và sau đó nhấn *Next*. Cửa sổ *Chart Wizard* tiếp theo trình bày các mẫu có sẵn cho kiểu đồ thị mà bạn chọn.

3.2.3. Chọn mẫu đồ thị

Khi bạn chọn kiểu đồ thị XY, cửa sổ được thể hiện ở hình 3.2 xuất hiện. Nó trình bày 5 mẫu chuẩn cho các đồ thị XY là:



- Đồ thị chỉ có các dấu hiệu điểm chấm thể hiện dữ liệu.

Hình 3.2: Bước thứ ba của Chart Wizard: chọn mẫu đồ thị XY.

- Đồ thị chỉ có các dấu hiệu điểm chấm thể hiện dữ liệu và đường nối chúng với nhau.

- Đồ thị chỉ có dấu hiệu điểm chấm thể hiện dữ liệu và lưới toạ độ.

- Đồ thị chỉ có dấu hiệu điểm chấm thể hiện dữ liệu và lưới bán lôgarit theo chiều thẳng đứng.

- Đồ thị chỉ có và các dấu hiệu lôgarit và các lưới lôgarit.

Nếu bạn chọn kiểu đồ thị 3-D Surface, cửa sổ như thể hiện trong hình 3.3 xuất hiện. Cửa sổ này trình bày bốn mẫu chuẩn đối với các đồ thị 3-D:

- Đồ thị dạng mặt 3-D.
- Đồ thị dạng khung đường ngang 3-D.
- Đồ thị dạng đường đồng mức có màu (hình chiếu xuống mặt bằng).
- Đồ thị dạng đường đồng mức (hình chiếu xuống mặt bằng).

Việc chọn mẫu cần phải tương tự với dạng đồ thị cuối cùng mà bạn mong muốn. Bạn có thể điều chỉnh hầu hết các thành phần đồ thị sau khi tạo đồ thị. Nhấn chuột vào núm *Next* để chuyển sang bước thứ tư.

| Chart Wizard - S | Step 1 of 4 - | Chart Type ?X |
|--|---------------|---|
| Standard Types | Custom Types | 1 |
| Chart type: Column Bar Line Pie XY (Scatter) Area Doughnut Radar Surface Bubble Stock | | Chart sub-type: |
| | | 3-D Surface. Shows trends in values across two dimensions in a continuous curve. Press and Hold to View Sample |
| 2 | Cancel | < Back <u>N</u> ext > <u>F</u> inish |

Hình 3.3: Bốn mẫu đồ thị 3-D Surface.

3.2.4. Xác định cách trình bày dữ liệu

Dữ liệu dối với kiểu đồ thị XY thường được sắp xếp trong các cột hoặc các hàng liền kể nhau. Nêu dữ liệu ở trong cột thì cột đầu tiên chứa dữ liệu x và các cột ở bên phải chứa dữ liệu y đối với một hoặc nhiều đổ thị. Dữ liệu y đối với mỗi đổ thị được biết đến dưới dạng một chuỗi dữ liệu và cũng có thể bao gồm dữ liệu x và một nhãn cho lời chú giải. Nếu thiếu dữ liệu x thì chúng ta sẽ sử dụng các giá trị 1, 2, 3, ...

Cửa số Chart Wizard thứ tự, hình 3.4, trình bày một đổ thị mẫu và yêu cấu thông tin về cách sắp xếp dữ liệu trên bảng tính. Với thông tin này, Chart Wizard kết hợp dữ liệu x và y và các nhãn tạo ra đổ thị. Đối với loại đổ thị XY, bạn phải suy nghĩ về 3 câu hỏi:

- Dữ hẹu được sắp xếp như thế nào, theo các cột hay hàng?

- Cột hoặc hàng đầu tiên có chứa dữ liệu x không, hay nó là chuỗi dữ liệu y đầu tiên?

- Ô đầu tiên trong mỗi hàng hoặc cột có chứa một nhãn cho chuỗi này không, hay nó chưa dữ liệu đầu tiên?

Sư sắp xếp của kiểu đồ thị 3-D hoàn toàn khác với kiểu đồ thị 2-D. Ở kiểu đồ thị 3-D, dữ liệu : tạo thành một vùng chữ nhật trên bảng tính. Hàng đầu tiên ở bên trên vùng chữ nhật này có thể chứa dữ liệu x, và cột đầu tiên ở bên trái có thể chứa dữ liệu y. Các giá trị dữ liệu x và y mà sử dụng cho một giá trị z đều là các giá trị mà ở trong cùng một hàng và cột như giá trị z. Nếu loại bỏ các giá trị y thì các nhãn s1, s2, s3... được sử dụng. Nếu loại bỏ các giá trị x thì các giá trị 1, 2, 3... được sử dụng như các nhãn.



Hình 3.4: Đô thị mẫu và các thông tin cần thiết.

Đối với đó thị 3-D, trong *Chart Wizard* bạn sẽ phải lựa chọn các tình huống tương tư như những câu hỏi dối với kiểu đồ thị XY:

- Dữ liệu đã được sắp xếp theo các hàng hay theo các cột? Sự lựa chọn này không tạo ra nhiều khác biệt trong kiểu đổ thị 3-D, nhưng đủ nhận ra dữ liệu x ở trong hàng đầu tiên hay trong cột đầu tiên.

- Cột hoặc hàng đầu tiên có chứa dữ liệu x không, hay nó là chuổi dữ liệu z đầu tiên?

- Cột hoặc hàng đầu tiên có chứa dữ liệu y không, hay nó là điểm dữ liệu đầu tiên trong mỗi chuối?

Bạn hãy nhân chuột vào các nút ở ngay sau câu trả lời đúng, sau đó nhân nút Next.

3.2.5. Bổ sung tiêu để

Cửa số Chart Wizard cuối cùng, thể hiện ở hình 3.5, sẽ cho phép bạn bố sung thêm một tiêu để, các tên của các trục toạ độ, và lời chú giải cho đổ thị. Ở các đổ thị XY, lời chú giải thể hiện tên chuỗi và các mẫu đường và dấu hiệu được sử dụng để trình bày nó. Sử dụng lời chú giải trên nhiều chuổi đổ thị XY để đánh dấu các chuối khác nhau. Ở đồ thị 3-D, lời chú giải thể hiện vùng các giá trị z mà dùng cho một màu trên đổ thị.

Kiểu văn bản mà bạn muốn đối với tiêu để và nhãn của các trúc ở trong hộp văn bản. Nhấn OK để kết thức buộc Chart Wizard cuối cùng và đạt đổ thị trên bảng tính.



Hình 3.5: Bước thứ 5 của Chart Wizard nhập vào loc chủ giải và các tiêu để.

3.3. VẼ ĐỒ THỊ MỘT BẢNG KỸ THUẬT

Ví dụ dưới đây sẽ làm rõ cách tạo ra các đồ thị của hàm số kỹ thuật. Đầu tiên chúng ta tính bảng các giá trị, và sau đó tạo đồ thị XY của các giá trị đó. Giả sử việc lập bảng đã làm xong như trong bảng 3.1.

| Tuổi bê tông | Cường độ bê tông dầm số 1 | Cường độ bê tông dầm số 2 | Cường độ bê tông cột số 1 |
|--------------|------------------------------|------------------------------|------------------------------|
| (ngay) | (MPa) | (MPa) | (MPa) |
| х | y1 | y2 | y3 |
| 1 | 10 | 13 | 20 |
| 3 | 12 | 17 | 36 |
| 5 | 15 | 22 | 38 |
| 7 | 18 | 24 | 41 |
| 14 | 20 | 29 | 48 |
| 28 | 22 | 30 | 51 |
| 56 | 23 | 32 | 53 |

Bảng 3.1. Các số liệu cường độ bê tông tăng theo thời gian

Bây giờ căn cứ vào bảng số liệu này chúng ta sẽ vẽ các đồ thị biểu diễn mối quan hệ giữa 4 đại lượng đã nói trong bảng. Trước tiên hãy gõ các số liệu vào 4 cột từ A đến D trong một bảng tính Excel mới mở. Sau đó hãy đánh dấu vùng dữ liệu bằng cách rê chuột bôi đen toàn bộ các dòng và cột đang chứa dữ liệu. Trình tự vẽ như sau:

- Nhấn chuột vào dấu hiệu vẽ đồ thị trên bảng chọn của Excel.

- Chọn dạng đồ thị XY.

- Chọn kiểu đồ thị có đường cong nối các điểm chấm dữ liệu. Bấm Next, cửa sổ như hình 3.6a sẽ hiện lên. Hãy bấm Next.

- Khi đó cửa sổ như hình 3.6b sẽ hiện ra.

Trên cửa sổ này có 5 phiếu mà bạn cần xem xét kỹ. Nội dung các phiếu đó như sau:

+ Title: để biên tập tên đồ thị và tên các trục toạ độ, đơn vị đo (xem hình 3.6b).

+ Axex: để biên tập các giá trị được ghi trên mỗi trục toạ độ (xem hình 3.6c).

+ Gridlines: để biên tập các đường dóng giá trị nằm ngang và thẳng đứng (hình 3.6d).

+ Legend: để biên tập vị trí ô chữ nhật chứa giải thích các ký hiệu trên đường cong (hình 3.6e).

+ Data Labels: để biên tập các giá trị ghi trên mỗi đường cong của đồ thị (hình 3.6g).



Hình 3.6a: Bước thứ 2 của Chart Wizzard.



Hình 3.6b: Biên tập tên đồ thị và tên các trục toạ độ.

| Chart Wizard - Step 3 of 4, | Chart Options |
|--|--|
| Titles Axes Gridlines Primary axis ✓ Value (X) axis ← ← ← ✓ ✓ Value (Y) axis | Legend Data Lahels Do thi $f = \frac{60}{50}$ $f = \frac{60}{40}$ $g = \frac{60}{20}$ $g = \frac{60}{40}$ $g = \frac{60}{$ |
| <u></u> | Cancel < <u>B</u> ack <u>N</u> ext > <u>F</u> inish |

Hình 3.6c: Biển tập các giả trị được ghi trẻa môi trạc toạ độ.

| Chart Wizard - Ste | p 3 of 4 - Chart Opt | tions | 199. 423 |
|---|--|-------------------|--|
| Titles Axes Value (X) axis Major gridlines Minor gridlines Value (Y) axis Major gridlines Minor gridlines | Gridlines Legend 60 50 50 50 50 50 50 50 50 50 50 50 50 50 | Data Labels Do ti | hi Cróng Đo bế trang dỹm sẽ 1 (MPs) y1 Crông Đó bế trang dỹm sẽ 2 (MPs) y2 Crông Đó bế trang cết sẽ 1 (MPs) y3 60 |
| 2 | Canœl | < <u>B</u> ad. | Next > <u>F</u> inish |

Hình 3.6d: Biên tập các đượng đóng giả trị nằm ngang và thẳng đứng.

| Chart Wizard - Step 3 of 4 | - Chart Options | |
|--|--------------------------------------|--|
| Titles Axes Gridlines | Legend Data Labels | T |
| Show legend Placement Bottom Corner Top Bight Left | Do thi |)é břitang 1 (MPa))é břitang 2 (MPa))é břitang (MPa) y3 |
| | Cancel < <u>B</u> ack <u>N</u> ext > | Finish |

Hinh 3.6e: Bien táp vi trí ó chứa giải thích các ký hiệu trên đường cong.



Hình 3.6g: Biên tập các giả trị ghi trên mỗi đường cong của đồ thị.

3.4. CHÌNH SỬA CÁC ĐỒ THỊ

Nhắc lại giả thiết là bạn đang dùng Excel-97. Để chỉnh sửa một đồ thị đã cài xong trên một trang tính nào đó, hãy nhấn đúp chuột trên đồ thị đó để làm xuất hiện cửa sổ bao quanh đồ thị. Khi vào trong cửa sổ đồ thị như trên hình 3.7, bạn có thể chọn các phần của đồ thị và sửa chúng một cách riêng rẽ, như sau:

- Nhấn chuột vào bên trong đường viền cửa sổ đồ thị để thay đổi định dạng của cửa sổ này. Khi đó sẽ hiện ra hộp thoại có tên là "Format Chart Area".

- Nhấn chuột trên trục toạ độ bất kỳ để thay đổi kiểu, các giới hạn, và tên của trục đó. Khi đó sẽ hiện ra hộp thoại có tên là " *Format Axis* ".

- Nhấn chuột trên chính đồ thị để thay đổi tất cả các kiểu văn bản và các kiểu mẫu cũng như các màu nền và màu cận cảnh. Khi đó sẽ hiện ra hộp thoại "*Format Plot Area*".

Nhấn chuột trên dòng kẻ gióng ngang của đồ thị để thay đổi kiểu đường gióng đó. Khi đó sẽ hiện ra hộp thoại có tên là " *Format Gridlines* ".



Hình 3.7: Bảng dữ liệu và đồ thị trên cùng một trang tính.

3.4.1. Thay đổi các đặc điểm của cửa sổ viền đồ thị

Nhấn chuột bên trong đường viền cửa sổ đồ thị để thay đổi định dạng của cửa sổ này. Khi đó sẽ hiện ra hộp thoại có tên là "*Format Chart Area*". Bạn có 3 phiếu để sử dụng là:

Pattern, Font, Properties.

Với phiếu Pattern bạn có thể sửa các đặc điểm sau:

 Làm cho khung cửa sổ bao quang đồ thị có bóng hoặc không có bóng, có 4 góc được vuốt tròn hoặc để góc vuông.

- Chọn mầu tuỳ ý theo một bảng mầu mẫu cho đường viền khung cửa sổ đó.

- Chọn mầu tô nền bên trong khung cửa sổ.

- Nếu bạn không muốn rắc rối thì cứ nhấn chuột chọn dấu chấm tròn vào mục Automatic trong các hộp thoại hiện trên màn hình. Nhớ đánh dấu \checkmark vào mục AutoScale.

Còn với phiếu Properties bạn có thể chọn chỉ một trong các đặc điểm sau:

- Cho phép di chuyển hay định lại kích thước các ô (bạn nên chọn khả năng này).
- Cho phép di chuyển nhưng không cho phép định lại kích thước các ô.
- Không cho phép di chuyển và cũng không cho phép định lại kích thước các ô.

3.4.2. Sửa đổi về các trục toạ độ trên đồ thị

Nhấn chuột trên trục toạ độ bất kỳ để thay đổi kiểu, các giới hạn, và tên của trục đó. Khi đó sẽ hiện ra hộp thoại có tên là "*Format Axis*". Bạn có 5 phiếu để sử dụng là:

Pattern, Scale, Font, Number, Aligment.

Sau đây là giải thích về cách sử dụng từng phiếu nói trên:

- Khi vào phiếu Pattern bạn nên chọn đặc tính Automatic cho nhanh.

- Khi vào phiếu Scale (Tỷ lệ xích vẽ đồ thị) bạn có thể chọn các đặc điểm sau:

+ chọn điểm giao của 2 trục toạ độ. Ví dụ bạn nên để gốc là 0,0.

+ chọn số lớn nhất và số nhỏ nhất trên trục đó.

- + chọn cự ly giữa các điểm đánh dấu trên trục toạ độ đó.
- + chọn đơn vị đo vẽ lớn nhất, đơn vị đo vẽ nhỏ nhất.
- + chọn vẽ theo thang đo lôgarit.
- Khi vào phiếu Font bạn có thể sửa các đặc điểm sau:
- + chọn font chữ cho các ghi chú trên trục toạ độ. Ví dụ bạn chọn font chữ Arial.
- + chọn kiểu font chữ (Font Style). Ví dụ bạn chọn Regular hoặc Italic.
- + chọn cỡ chữ (Size). Ví dụ bạn chọn cỡ chữ 10.
- + chọn: có hoặc không có gạch dưới các chữ.
- + chọn mầu sắc của chữ. Ví dụ bạn chọn "Automatic".
- + chọn mầu nền của chữ. Ví dụ bạn chọn "Automatic".
- Khi vào phiếu Number bạn có thể chọn định dạng thể hiện các dữ liệu số trên đồ thị:
- + Ví dụ bạn chọn kiểu General hoặc kiểu Sientific, kiểu Number.
- + Nếu tính ngày thì chọn kiểu Date. Nếu tính tiền thì chọn kiểu Currency, v.v...

- Khi vào phiếu Alignment bạn có thể chọn định hướng cho dòng chữ thể hiện trên trục toạ độ.

+ Ví dụ bạn nên chọn kiểu Automatic là thông dụng nhất.

3.4.3. Định dạng đồ thị

Nhấn chuột trên chính đồ thị để thay đổi kiểu mẫu cũng như các màu nền và màu cận cảnh. Khi đó sẽ hiện ra hộp thoại có tên là "*Format Plot Area*". Bạn có 1 phiếu để sử dụng là *Pattern*. Nội dung chỉnh sửa cũng giống như chỉnh sửa *Pattern* đã giới thiệu ở mục 3.4.1.

3.4.4. Định dạng các đường dóng ngang trên đồ thị

Nhấn chuột trên dòng kẻ dóng ngang của đồ thị để thay đổi kiểu đường dóng đó. Khi đó sẽ hiện ra hộp thoại có tên là "*Format Gridlines*". Bạn có 2 phiếu để sử dụng là *Pattern* và *Scale*. Nội dung chỉnh sửa cũng giống như chỉnh sửa *Pattern* và mục *Scale* đã giới thiệu ở mục 3.4.2.

3.4.5. Bổ sung thêm các đường cong cho một đồ thị

Bạn đọc cần lưu ý là Excel-97 có những tính năng thông minh. Ví dụ ngay trên hình 3.8, cột dữ liệu X đã được Excel tự động tô đường viền mầu xanh để tách riêng nó ra so với 3 cột còn lại chứa các giá trị Y1, Y2 và Y3. Như thế Excel-97 đã tự động hiểu rằng X là biến số con; Y1, Y2 và Y3 là những hàm số phụ thuộc X. Do đó Excel-97 sẽ lấy trục nằm ngang là trục của các giá trị của X, còn trục thẳng đứng là trục ghi các giá trị của các hàm số Y1, Y2 và Y3.

Nếu bạn muốn vẽ cả 4 đường cong lên cùng một đồ thị, nghĩa là trục hoành chỉ ghi số thứ tự của các dòng chứa số liệu chứ không ghi số liệu thì khi chọn kiểu đồ thị bạn không chọn kiểu XY mà phải chọn kiểu LINE như hình 3.8.



Hình 3.8: Ví dụ trường hợp chọn kiểu đồ thị LINE để vẽ cả 4 đường cong.

Các thao tác tiếp theo giống như đã trình bày ở mục 3.2. Kết quả cuối cùng sẽ như hình 3.9 dưới đây. Lưu ý là tuy số liệu giống như của đồ thị hình 3.8 nhưng lại vẽ ra 4 đường cong. Hơn nữa bạn có thể ghi số liệu ngay lên các điểm trên các đường cong đó để dễ theo dõi và phân tích.

Giả sử sau khi vẽ đồ thị xong bạn lại muốn thêm một cột số liệu nữa và muốn vẽ thèm đồ thị biểu diễn cột số liệu mới thêm vào này. Cách tốt nhất là đừng nghĩ đến việc chỉnh sửa mà bạn nên vẽ lại toàn bộ đồ thị mới vì thực ra cũng chỉ mất vài phút. Thông thường, một đồ thị thể hiện nội dung của các ô được liên kết với nó. Nếu nội đung của các ô thay đổi, đồ thị cũng thay đổi để phản ánh giá trị mới.

3.5. TẠO RA CÁC ĐỒ THỊ LÔGARIT VÀ BÁN LÔGARIT

Kiểu đồ thị LINE có thể có các trục toạ độ với đơn vị đo tuyến tính hoặc đơn vị đo lôgarit. Giả sử bạn đang có một đồ thị với các đơn vị đo trên trục toạ độ là đơn vị đo tuyến tính. Nếu bạn muốn chuyển đơn vị đo các trục từ đơn vị đo tuyến tính sang đơn vị đo lôgarit, hãy nháy chuột vào trục toạ độ mà bạn muốn chuyển đơn vị, trên màn hình sẽ xuất hiện cửa sổ *Format Axis*. Hãy chọn phiếu *Scale* rồi dánh dấu vào hộp kiểm tra mục *Logarithmic scale*. Trong ví dụ sau đây sẽ minh hoạ quá trình tính một phương trình rồi vẽ nó dưới dạng đồ thị lôgarit.



Hình 3.9: Kết quả vẽ đồ thị kiểu LINE với 4 đường cong theo 4 cột số liệu.

3.6. TẠO ĐỒ THỊ KHÔNG GIAN BA CHIỀU

Tất cả các đồ thị ba chiều của Excel đều có các đường kẻ ô được đặt cách đều nhau theo cả phương x và y, để việc kẻ ô là đều nhau cho dù bất cứ giá trị nào mà bạn đưa ra cho các trục x và y. Trong một số trường hợp, việc kẻ ô đều nhau này có thể là một bài toán, chẳng hạn như khi việc kẻ ô làm méo mó các kết quả. Tuy nhiên, bằng cách vẽ liên tiếp những lát cắt qua dữ liệu và dịch chuyển chúng với lượng thích hợp, bạn có thể tạo ra một đồ thị khung dây ba chiều sử dụng kiểu đồ thị hai chiều XY.

Trước tiên bạn cần xác định một hệ toạ độ vuông góc u, v và w. Vẽ trục x nằm ngang, trục v với một góc ϕ , và trục w theo chiều thẳng đứng. Tiếp theo, bạn thể hiện ba trục này trên mặt phẳng hai chiều, với các phương trình biến đổi này: $x = u + v\cos\phi$ và y = w+ vsin ϕ . Để vẽ đồ thị điểm dữ liệu ba chiều (u, v, w) hãy đưa nó vào trong các phương trình biến đổi và tính điểm dữ liệu hai chiều (x, y).

Để tạo bảng tính, bạn cần phải đưa dữ liệu ba chiều vào trong ba cột để bạn có thể trình bày chúng trong hai cột mà sau đó có thể được vẽ trên đồ thị XY. Dữ liệu đối với kiểu đồ thị này thường xuất hiện dưới dạng các lát cắt qua miền được vẽ đồ thị. Mỗi lát cắt đi từ một biên này tới biên khác. Bạn xếp chồng các tệp dữ liệu này, sao cho điểm dữ liệu đầu tiên của một mặt cắt theo sau điểm dữ liệu cuối cùng của mặt cắt trước.

Biểu diễn đồ thị của bài toán là một đường sẽ được vẽ (kéo) từ điểm dữ liệu cuối cùng của mỗi mặt cắt tới điểm dữ liệu đầu tiên của mặt cắt tiếp sau. Các đường này cắt ngang qua phần giữa của đồ thị, điều đó thì rõ ràng bạn không muốn.

Nhưng bạn có thể ngăn không cho Excel vẽ các đường nguợc lại bằng cách chèn một hàng trống vào giữa mỗi mặt cắt của dữ liệu. Excel sẽ coi dòng trống như một điểm dữ liệu không có trong phần giữa của một miền dữ liệu, và sẽ không vẽ vật đánh dấu hay các đường đến các điểm đó.

3.7. VË HÌNH

Cùng với các công cụ vẽ đồ thị, Excel cũng đưa ra các công cụ để vẽ các hình. Nếu kích hoạt thanh công cụ vẽ, bạn có thể vẽ các đường, các vòng tròn và các hộp bằng các màu khác nhau để làm rõ nét các kết quả và các phép tính khác nhau trên các bảng tính. Để hiển thị thanh công cụ vẽ, sử dụng lệnh *Toolbar* trên bảng chọn *Options*. Thanh công cụ vẽ chứa các công cụ thể hiện như trong hình 3.10.

| | | | | | n ya pani shipan niya di sa ay shi s | | | | . <u> </u> | • • • | | | |
|----------|---|------------|-----|---|--|-----|-------|----|------------|-------|------|----------|---|
| Decenti, | 3 | f. | • | | | 194 | 1.19. | | | | | <u>,</u> | 2 |
| Draw - | ß | AutoShapes | · \ | X | ₽ 4 | 1 🗘 | 3. | .# | A | • = | | | Ø |

Hình 3.10: thanh công cụ vẽ

Hơn nữa, bạn có thể sử dụng khả năng vẽ đồ thị dữ liệu của Excel để vẽ các hình phức tạp trên màn hình như việc vẽ đường được cắt từng đoạn được tạo ra với xấp xỉ 425 cặp dữ liệu mà hình thành nên 53 đoạn đường gián đoạn.

Việc vẽ trên đồ thị có thể giúp minh hoạ chức năng đang được vẽ. Ví dụ. nếu bạn đang vẽ đồ thị đầu ra của mạch điện tử, vẽ mạch đơn giản ở góc của đồ thị làm cho các kết quả ý nghĩa hơn đối với đầu đọc.

Thật không may, các công cụ vẽ của Excel làm việc chỉ trên các bảng tính, không phải trên các bảng đồ thị. Bạn có thể tạo ra một đồ thị được gắn trong bảng tính và sử dụng các công cụ vẽ để thêm một biểu đồ, hoặc bạn có thể vẽ trên một bảng tính đồ thị bằng cách sử dụng các đoạn đường.

Chương 4

SỬ DỤNG "MACRO" VÀ "VISUAL BASIC FOR APPLICATIONS"

Các Macro làm tăng lên rất nhiều sức mạnh của Excel. Với các Macro, chúng ta có thể tự động hóa hầu hết các công việc lặp đi lặp lại.

Bạn cũng có thể tạo ra các hàm thông thường để sử dụng trong bảng tính, cũng như các chương trình Macro đầy đủ để tự động hóa các công việc phức tạp.

Bộ ghi Macro của Excel ghi từng bước mà người sử dụng làm dưới dạng một danh sách các lệnh Macro. Bạn có thể soạn thảo danh sách lệnh này thành chương trình Macro. Đối với các chương trình Macro tìm sai, Excel bao gồm chương trình bổ sung Macro Debugger.

Chương trình bổ sung bao gồm nhiều khả năng tìm sai bậc cao, bao gồm các điểm ngắt, các điểm vệt và sự hiển thị biến.

Visual Basic làm tăng đáng kể sức mạnh của Excel hơn so với khi dùng các Macro thông thường. Với thủ tục của Visual Basic, bạn có thể tự động hoá hầu hết các nhiệm vụ bảng tính lặp đi lặp lại. Bạn cũng có thể tạo các hàm khách hàng để sử dụng trong bảng tính, cũng như hoàn thiện các chương trình để tự động hoá các nhiệm vụ phức tạp.

Bộ ghi Macro của Excel ghi mỗi hành động bạn hiểu như một danh sách các lệnh Visual Basic. Bạn có thể soạn thảo danh sách các lệnh này vào trong chương trình Macro mà bạn có thể sử dụng. Để gỡ rối chương trình, Excel gộp cửa sổ Debug mà biểu diễn các giá trị của các biến và cho phép người sử dụng xem chương trình đang thực hiện, mỗi lần một bước.

Visual Basic có nhiều khả năng hơn nhiều so với những khả năng đã thể hiện trong chương trình ngắn này. Như trước đây đã đề cập, tất cả các cuốn sách của tác giả chỉ viết về Visual Basic, do vậy có được một trong số những cuốn sách đó để thực sự hiểu các khả năng của ngôn ngữ.

Nói chung, Macro có mục tiêu đơn giản là thực hiện việc lặp lại một cách tự động toàn bộ quá trình gõ phím để tiết kiệm thời gian cho bạn khi thực hiện các hành động lặp đi lặp lại. Tuy nhiên các tác giả của Excel đã phát triển Macro dần thành một ngôn ngữ lập trình đầy đủ. Macro Excel không những là sự tự động lặp lại các thao tác giống như đánh phím lần đầu, mà chúng còn được sử dụng để tạo ra các hàm bảng tính thông thường hoặc thậm chí hoàn thành các chương trình máy tính. Ngôn ngữ Macro của Excel

bao gồm phần lớn các hàm và cấu trúc điều khiển, người sử dụng sẽ mong đợi tìm trong ngôn ngữ bậc cao.

Với một trình độ cao hơn nhiều nữa, Visual Basic của Excel còn làm thêm được nhiều việc kỳ diệu hơn khiến cho có thể nói nếu một kỹ sư nắm được ngôn ngữ này thì chỉ cần dùng Excel cũng hoàn thành được phần lớn các công tác tính toán thực hành của mình.

4.1. SỬ DỤNG CÁC LỆNH MACRO

Các Macro Excel được đặt trên Macro riêng rẽ, mà xử lý hoàn toàn khác với bảng tính. Trên một bảng tính, các ô được tính toán theo trật tự tính lại tự nhiên, mà trong đó các ô trước của một ô (những ô mà một ô phụ thuộc vào) luôn được tính toán trước khi một ô được tính, cho dù ô trước được đặt trên bảng tính. Trên bảng Macro, sự tính toán tiếp tục xuống một cột, một ô kế tiếp nhau, thực hiện mỗi ô theo trật tự.

Tất cả các lệnh Macro là các hàm; có nghĩa là, chúng luôn khai báo một giá trị. Các hàm Macro nhận thao tác thường khai báo TRUE hoặc FALSE, phụ thuộc vào các số liệu thao tác này chúng được cho là để thực hiện có xảy ra không. Hầu hết các lệnh Macro không thể sử dụng trên bảng tính, nhưng tất cả các lệnh bảng tính có thể được sử dụng trên bảng Macro.

Hầu hết các lệnh Macro tương đương lệnh được đặt tên cho lệnh này, hoặc lệnh và bảng chọn mà chúng cư trú. Chẳng hạn như, hàm FORMAT.NUMBER thực hiện lệnh *Number* trên bảng chọn *Format*.

Chương này mô tả một số lệnh Macro hữu ích. Bạn đọc nào muốn tìm hiểu kỹ hơn nên xem cuốn sách "*Microsoft EXCEL Function Reference*", nó bao gồm chương trình chuyên dùng Excel cho sự mô tả đầy đủ về mỗi lệnh Macro.

4.2. TẠO MACRO LỆNH ĐỂ TỰ ĐỘNG HÓA CÁC BẢNG TÍNH

Tập hợp lệnh Macro cơ bản nhất là để tự động hóa các hành động lặp lại trên một bảng tính. Các Macro của loại này được biết đến như các Macro lệnh, vì chúng được thực hiện giống như các lệnh bảng chọn. Hầu hết mọi công việc lặp lại có thể do một Macro thực hiện. Lợi ích chủ yếu đối với những Macro này là tiết kiệm thời gian cho bạn khi bạn thực hiện các hành động lặp lại và tạo ra các bảng tính tự động cho những người mà không phải là chuyên gia về Excel. Chúng cũng có ích bởi vì các bảng tính dường như tự tạo ra không cần bạn chạm vào bàn phím.

Các Macro lệnh đều đơn giản để tạo ra và thậm chí đơn giản hơn để sử dụng. Để tạo ra một Macro lệnh, hãy thiết lập một bảng tính bình thường như trước khi Macro này được thực hiện, và chọn lệnh *Record* trên bảng chọn *Macro*. Hộp thoại *Record Macro* xuất hiện, như thể hiện trong hình 4.1.

Trong hộp thoại này, gõ tên cho Macro của bạn và tên đó không bắt buộc, ấn phín: được sử dụng trong tổ hợp với phím Ctrl như lệnh tắt để quay lại Macro này. Bạn không cần chỉ rõ phím lệnh tắt, nhưng nó sẽ cho phép bạn thực hiện một Macro nhanh hơn. Hộp thoại này cũng cho phép chúng ta lựa chọn giữa việc ghi Macro trên bảng Macro hoặc trên toàn bộ bảng Macro. Đặt các Macro mà bạn sử dụng một cách thường xuyên trên toàn bộ bảng Macro, vì nó được mở bất cứ khi nào bạn chạy Excel. Đặt các Macro thiết kế riêng trên các bảng Macro thường lệ, mà phải được mở trước khi các Macro trên chúng có thể được sử dụng. Sau đó mở bảng Macro thích hợp khi bạn đang làm việc trên thiết kế được kết hợp.

| Record Macro | | ?× |
|------------------------------------|--|----------|
| Macro name: | | |
| Shortout key: | Store macro <u>i</u> n: This Workbook | - |
| Description: Macro recorded 3/8 | 3/2004 by bm | |
| ~ | ОК | Cancel |

Hình 4.1: Hộp thoại Recorder Macro.

Ngay sau khi bạn nhấn chuột vào OK thì Excel bắt đầu ghi nhớ các thao tác của bạn. Tất cả mọi thứ bạn làm được ghi trên bảng Macro ví dụ như các thao tác mở và đóng các tệp, gõ các công thức, và v.v... Khi bạn đã kết thúc mọi việc, hãy chọn lệnh stop recorder trên bảng chọn Macro.

Nếu bạn đã ghi Macro của mình trên một bảng Macro thường lệ thì hãy chuyển sang bảng Macro sử dụng bảng chọn Window, sẽ nhìn thấy Macro mới của bạn. Nếu Macro của bạn ở trên toàn bộ bảng Macro, thì sử dụng lệnh *Unhide* trên bảng chọn Window để làm cho nó có thể nhìn thấy được.

Để ghi Macro thứ hai trên bảng Macro, hãy chọn ô trên bảng Macro nơi mà bạn muốn bắt đầu ghi và chọn lệnh set recorder trên bảng chọn Macro. Sau đó sử dụng các lệnh start recorder và stop recorder trên bảng chọn Macro để bắt đầu và dừng việc ghi. Khi đã kết thúc việc ghi, hãy chọn lại ô đầu tiên của Macro một lần nữa và sử dụng lệnh Define Name trên bảng chọn Formula để đặt tên cho Macro này, thiết lập kiểu của nó và gản một phím lệnh tắt.

Khi bạn đang ghi các Macro, các tham chiêu ô được ghi theo chế độ tuyệt đối, chế độ này làm cho chúng truy nhập cùng các ô như nhau mỗi lần. Nhưng một vài Macro đòi hỏi các tham chiếu tương đối. Chẳng tạn như, bạn muốn một Macro mà định dạng

một ô để định dạng mà bạn ở trong đó, không phải ô mà bạn đã ở trong đó khi bạn đã ghi Macro này. Để đổi sang việc ghi theo chế độ *relative*, chọn lệnh *relative record* trên bảng chọn Macro. Sử dụng lệnh *Absolute Record* trên bảng chọn Menu để thay đổi trở lại.

Thay vì việc cho Macro một phím lệnh tắt hoặc thực hiện nó từ bảng chọn Macro, bạn có thể gán một Macro cho một đối tượng, như là một nút, và thực hiện Macro này bằng cách nhấn chuột vào đối tượng đó. Bạn có thể tạo ra một nút bằng công cụ nút trên thanh công cụ hữu dụng, nhưng đối tượng nào đó trên bảng tính có thể được gắn vào một Macro, bao gồm đồ thị được gắn vào hoặc môi trường vẽ, như hình chữ nhật hoặc đường kẻ. Để gán một Macro cho một đối tượng, chọn Macro được gán từ hộp thoại, và nhấn chuột vào OK.

Để chạy một Macro lệnh, giữ Ctrl và ấn phím lệnh tắt của nó (nếu bạn đã gán một phím lệnh tắt), nhấn vào đối tượng của nó (nếu bạn đã gán một Macro cho một đối tượng), hoặc chọn Run từ bảng chọn Macro và chọn tên của nó từ hộp thoại Run Macro.

4.2.1. Macro định dạng ô

Bây giờ chúng ta sẽ thực hành một ví dụ để tạo một Macro lệnh có nhiệm vụ định dạng và vẽ các ô. Đây là một Macro đơn giản mà định dạng các nội dung của ô đã chọn hoặc các ô là 0.00E+00 và vẽ một hộp xung quanh những ô này. Gắn Macro cho một nút trên bảng tính.

Hãy thực hiện lần lượt các thao tác sau:

1. Mở một bảng tính mới, gõ một chữ số trong một ô để định dạng trong khi ghi Macro.

2. Chọn lệnh Record trên bảng chọn Macro.

3. Trong hộp thoại *Record Macro*, đặt tên Macro là *Formatlt*, thiết đặt phím lệnh tắt là r, và nhấn chuột vào OK để bắt đầu ghi.

4. Chọn lệnh *Relative Record* trên bảng chọn Macro nếu nó xuất hiện. Nếu lệnh *Absolute Record* là ở trên bảng chọn, máy ghi đã ở trong chế độ Relative.

5. Nhấn chuột vào ô chứa chữ số này, chọn lệnh Number trên bảng chọn Format, chọn định dạng 0.00E+00, và nhấn OK.

6. Chọn lệnh Border trên bảng chọn Format, nhấn vào Outline, và nhấn OK.

7. Chọn lệnh Stop Recorder trên bảng chọn Macro.

8. Chọn bảng Macro từ bảng chọn Window.

Macro xuất hiện trên bảng Macro, như thể hiện trong hình 4.2. Chú ý rằng RETURN () ở cuối của Macro. Tất cả các Macro phải kết thúc với hàm RETURN để đưa ra điều khiển cho người sử dụng. Không có RETURN, Excel tiếp tục thực hiện các ô trong các cột cho đến khi nó không thể tìm thấy cái gì nữa.

68

Nếu xem Macro này một cách cẩn thận, bạn sẽ thấy rằng nó không đúng như cái bạn muốn. Nó có hàm SELECT trong dòng 2, mà được viết khi bạn đã chọn ô để định dạng. Vì muốn Macro này định dạng bất cứ ô nào đã chọn và không di chuyển tới một vài ô khác, bạn phải bỏ hàm này.

9. Chọn ô A2 và sử dụng lệnh *Delete* trên bảng chọn *Edit* để xóa ô này và di chuyển tới các ô bên dưới nó.

10. Chọn bảng tính từ bảng chọn Window.

11. Chọn Toolbar trên bảng chọn Options và chọn Utility để có thể hiển thị các thanh công cụ hữu dụng.

| Macro | | | ?X |
|----------------|--------------------|----------|-----------|
| Macro name: | | | |
| | | 3 | Run |
| | | 1 | Cancel |
| | | _ | Step Into |
| | | - | Edit |
| | | | Create |
| | | <u> </u> | Delete |
| Macros in: | All Open Workbooks | - | Options |
| Description —- | | | |
| | | | |

Hình 4.2: Macro Formatlt được ghi trên bảng chọn Macro.

12. Nhấn vào công cụ nút (biểu tượng trông giống như hình chữ nhật màu xám có các góc tròn và một vết tối) và vẽ một nút trên bảng tính bằng cách nhấn và kéo vào nơi thuận tiện.

13. Chọn lệnh Assign to Object trên bảng chọn Macro. Trong hộp thoại này, chọn Macro FormatIt và nhấn OK.

14. Sử dụng trỏ chuột để chọn văn bản trên bề mặt nút này và gõ Formatlt.

Lúc này Macro đã sẵn sàng dể sử dụng. Gõ một vài chữ số trong một ô và nhấn vào nút *Formatlt* để xem điều gì xảy ra.

Khi đã được tạo ra, bảng Macro được lưu giữ hoặc được mở giống như bảng tính. Để sử dụng một Macro, bảng Macro mà chứa nó phải mở. Những Macro bạn tạo ra có thể di chuyển được; bạn có thể sử dụng những Macro trên bảng Macro mở với bảng tính mở nào đó, không chỉ một bảng tính đã kích hoạt khi chúng được tạo ra.

4.3. LẬP TRÌNH VỚI MACRO

Các Macro không cần đơn giản như Macro ở ví dụ trước. Bạn có thể viết các chương trình đầy đủ bao gồm các hộp thoại và bảng chọn thông thường, những phép tính lặp, sự truy nhập và tạo tệp, và sự điều khiển thông thường của bảng tính. Nhược điểm chính của Excel là tốc độ của nó. Vì nó là một ngôn ngữ được dịch cho nên nó không chạy nhanh, mặc dù nó nhanh hơn đủ đối với các ứng dụng đơn giản nhất. Nếu bạn có một hàm thông thường cần tốc độ của ngôn ngữ biên soạn, hãy tạo ra nó dưới dạng một modul trong Dynamic Link Library (DLL) (nguồn CODE trên Macintosh) và sau đó sử dụng hàm CALL hoặc REGISTER để truy nhập nó.

4.3.1. Bố trí các Macro

Thông thường (nhưng không bắt buộc) người ta bố trí các Macro trong khuôn dạng ba cột sau đây:

- Cột thứ nhất được sử dụng để dán nhãn các ô đã đặt tên trong Macro. Trong cột này, đưa vào tên cho mỗi ô trong cột thứ hai mà bạn muốn đặt tên. Những cột này bao gồm tên Macro và bất cứ ô nào đã đặt tên được sử dụng để chứa dữ liệu.

- Cột thứ hai chứa Macro và những vùng đã đặt tên để chứa dữ liệu.

- Cột thứ ba dành cho lời chú giải.

4.3.2. Sự thao tác các tham chiếu ô

Vì ngôn ngữ Macro được thiết kế để thao tác các bảng tính, nên nhiều lệnh trở lại các tham chiếu ô. Hãy cẩn thận khi thiết kế một chương trình để thao tác những tham chiếu. Nếu bạn chứa một tham chiếu trong một ô, thì giá trị của ô đó sẽ là những nội dung của tham chiếu, chứ không phải tham chiếu của chính nó. Vấn đề thường xảy ra nhất khi bạn có một hàm mà chấp nhận một tham chiếu như là một đối số, bạn sử dụng hàm khác trở lại tham chiếu như là đối số đó. Chẳng hạn bạn đang sử dụng hai hàm, FUNCTIONA () và FUNCTIONB (). FUNCTIONA () đòi hỏi một tham chiếu cho một đối số, và FUNCTIONB () khai báo tham chiếu.

Nếu bạn đặt công thức này vào một ô:

```
= FUNCTIONA(FUNCTIONB())
```

nó làm việc như được mong đợi. Tuy nhiên, nếu bạn lưu giữ kết quả của FUNCTIONB () trong một ô, như trong A1, và sau đó sử dụng giá trị đó như một đối số đối với FUNCTIONA() trong một ô khác, như trong công thức:

= FUNCTIONA(A1)

sẽ không làm việc, bởi vì ô A1 chứa nội dung của tham chiếu, chứ không phải tham chiếu.

Nếu bạn phải lưu giữ một tham chiếu trong một ô cho sự thao tác thêm nữa, trước tiên hãy chuyển đổi nó thành một chuỗi và lưu giữ chuỗi đó, còn hơn là tham chiếu thực. Nhiều hàm đã trở lại tham chiếu như một chuỗi. Đối với những hàm mà nó không trở lại, sử dụng hàm REFTEXT để biến đổi tham chiếu này thành vãn bản. Khi bạn cần sử dụng tham chiếu này, biến đổi ngược trở lại thành tham chiếu bằng hàm TEXTREF. Chẳng hạn như, nếu bạn đã nhập công thức này vào ô A1:

= REFTEXT(FUNCTIONB())

và công thức này vào ô A2:

= FUNCTIONA(TEXTREF(A1))

tham chiếu này sẽ làm việc một cách chính xác.

Khi bạn chuyển đổi một tham chiếu thành văn bản, nó hầu như luôn luôn ở trong kiểu tham chiếu ô R1C1. Ở kiểu R1C1, vị trí của một ô là theo số hàng và số cột của nó thay vì kiểu chữ và số. Chẳng hạn như, R7C3 nghĩa là hàng 7, cột 3, đó là ô C7. Hãy nhớ điều này nếu bạn dự định thao tác một địa chỉ ô.

4.3.3. Các biến bảng Macro

Một bảng Macro có thể chứa hai kiểu biến: các giá trị và ô được đặt tên tại thời điểm thiết kế bằng lệnh *Define Name* trên bảng chọn *Formula*, hoặc bằng hàm SET.NAME trong Macro thực hiện. Chẳng hạn như, hàm này:

= SET.NAME ("theLength", 5)

định nghĩa tên *theLength* dưới dạng số 5. Nếu bạn sử dụng tham chiếu ô như đối số thứ hai, thì tên này được định nghĩa là tham chiếu ô, chứ không phải là những nội dung của ô. Do vậy, khi bạn sử dụng tên này, thì nó được đánh giá như tham chiếu ô, mà đánh giá những nội dung hiện thời của ô.

Các ô có thể chứa các giá trị trên bảng Macro theo cùng cách thức như các ô trên một bảng tính chứa các giá trị. Khi một ô được tính toán, nó nhận giá trị của hàm nó chứa. Hơn nữa, giá trị ô có thể được thiết đặt từ một ô khác sử dụng hàm SET.VALUE. Ví dụ, những hàm này:

đặt giá trị của ô B7 trên bảng Macro là 6, và ô được đặt tên I tới giá trị (I + 1). Cả hai giá trị này tham chiếu tới các ô trên bảng Macro.

Để thay đổi một giá trị trên bảng tính, hãy sử dụng hàm FORMULA. Hàm FORMULA làm việc giống như việc gõ máy trên thanh công thức của bảng tính. Nó nhận một hoặc hai đối số. Đối số thứ nhất phải là văn bản và là cái bạn muốn được chèn vào ô. Đối số thứ hai là tham chiếu ô mà trong đó để chèn văn bản. Nếu bạn bỏ qua đối số thứ hai, ô được chọn hiện thời được sử dụng.

Xử lý đầu vào và đầu ra: Để nhận đầu vào từ người dùng, sử dụng hàm INPUT, hàm này hiển thị hộp thoại và các yêu cầu mà người sử dụng gõ cái gì đó. Để gửi một cách đơn giản cho người sử dụng một thông báo, hãy sử dụng hàm ALERT. Bạn có thể tạo ra những hộp thoại thông thường với Excel cho cả đầu vào và đầu ra. Những hộp thoại thông thường được thiết kế bằng Dialog Editor và được hiển thị bằng hàm DIALOG.BOX, như được mô tả trong phần sau trong chương này.

Bạn có thể truy nhập dữ liệu trong các tệp đĩa ASCII bằng các hàm FOPEN, FCLOSE, FREAD, FWRITE, FPOS, FREADLN, và FWRITELN. Cũng có thể nhận kích cỡ tệp bằng FSIZE. Những lệnh này cho phép bạn mở và đóng các tệp văn bản và để đọc, ghi dữ liệu.

4.3.4. Điều khiển lưu đồ chương trình

Bạn điều khiển dòng thao tác trong một Macro hoạt động bằng các lệnh vòng lặp, bước nhảy, các điều kiện, và các chương trình con. Các vòng lặp được điều khiển bằng các hàm FOR, FOR.CELL, WHILE và NEXT. Hàm FOR bắt đầu một vòng lặp đếm được trong đó bộ đếm tăng lên mỗi lần vòng lặp được thực hiện. Hàm FOR.CELL tương tự hàm FOR, nhưng các vòng lặp qua các ô trong tham chiếu ô thay vì việc táng một số. Hàm WHILE thực hiện lặp lại một vòng lặp cho đến khi điều kiện khai báo FALSE. Tất cả những vòng lặp này được kết thúc bằng hàm NEXT.

Macro của bạn có thể thực hiện bước nhảy vô điều kiện, tương tự như GOTO trong Fortran hoặc Basic, bằng hàm GOTO. Nhánh rẽ có điều kiện được thực hiện với hàm IF khi các hàm GOTO được sử dụng như các đối số của nó. Để gọi một chương trình Macro khác như một chương trình con, đơn giản đặt tên của nó trong Macro của bạn, được theo sau bởi các đối số của nó trong các dấu ngoặc đơn. Các chương trình con trở lại ô kế tiếp trong chương trình gọi khi chủng gặp hàm RETURN. Sử dụng hàm ARGUMENT để chỉ rõ nơi mà các tham số của chương trình con được lưu giữ.

4.3.5. Tạo ra các hộp thoại

Bạn sử dụng Dialog Editor để tạo ra những hộp thoại thông thường. Dialog Editor tạo ra một bảng định nghĩa hội thoại, mà chứa sự mô tả của mỗi khoản mục trên hộp thoại này.

Mỗi hàng của bảng định nghĩa hội thoại mô tả một khoản mục khác nhau. Dòng đầu tiên là đặc biệt, vì chính nó mô tả hộp thoại. Nếu bạn tính đến tên cho hộp thoại trong trường văn bản của hàng thứ nhất, hộp thoại này sẽ có thể di chuyển được; nếu không, nó sẽ cố định. Nếu bạn đặt tham chiếu tệp vào tệp Help trong cột thứ nhất của hàng một, tệp đó sẽ được mở như chủ đề Help nếu bạn thêm nút Help vào hộp thoại.

Cột đầu tiên của bảng định nghĩa hội thoại chứa một mã mà nhận biết kiểu khoản mục.

Các cột thứ hai và thứ ba chứa vị trí x-y của góc trên bên trái của khoản mục, được đo theo các điểm (một điểm bằng 1/72 in) từ góc trên bên trái của hộp thoại. Các cột thứ tư và thứ năm chứa chiều rộng và chiều cao của khoản mục được đo theo điểm. Cột thứ sáu chứa văn bản được gắn vào đối tượng. Văn bản là đầu đề của nút, các nội dung của hộp văn bản, hoặc nhãn của nút tuỳ chọn hoặc hộp kiểm tra.

Cột thứ bảy chứa giá trị ban đầu và kết quả cho mỗi mục. Khi người sử dụng lựa chọn trong hộp thoại và nhấn vào một nút để thoát ra, số khoản mục của nút được lựa chọn (đếm ngược từ đỉnh của bảng định nghĩa hội thoại) được khai báo bởi hàm DIALOG.BOX, và tất cả sự lựa chọn đã thực hiện được đưa ra trong cột thứ bảy của bảng định nghĩa hội thoại.

4.4. TẠO RA MACRO HÀM

Khi bạn định nghĩa một tên trên bảng Macro bằng lệnh *Define Name* trên bảng chọn *Formula*, bạn có sự lựa chọn về việc định nghĩa nó dưới dạng Macro lệnh hoặc Macro hàm. Như bạn đã thấy, Macro lệnh thường thực hiện một thao tác. Những Macro hàm tính toán và đưa ra một hoặc nhiều giá trị. Kiểu hàm Macro đơn giản nhất và phổ biến nhất khai báo một giá trị đơn. Tuy nhiên, những Macro lệnh có thể khai báo một chuỗi các giá trị, và chúng được xử lý theo cách thức tương tự như bất kỳ hàm nào khác mà khai báo một chuỗi.

| Mã | Мџс |
|----|---|
| 1 | Mặc định nút OK, được lựa chọn tự động nếu người sử dụng ấn Return |
| 2 | Nút Cancel |
| 3 | Nút OK |
| 4 | Mặc định nút Cancel, được lựa chọn tự động nếu người sử dụng ấn Return |
| 5 | Nhãn văn bản, văn bản tĩnh được sử dụng để dán nhãn cho các khoản mục trong hộp thoại |
| 6 | Hộp soạn thảo văn bản, được sử dụng để nhận văn bản từ người dùng |
| 7 | Hộp soạn thảo số nguyên, được sử dụng để nhận số nguyên từ người sử dụng |
| 8 | Hộp soạn thảo số, được sử dụng để nhận một chữ số từ người dùng |
| 9 | Hộp soạn thảo công thức, được sử dụng để nhận công thức từ người sử dụng |
| 10 | Hộp soạn thảo tham chiếu, được sử dụng để nhận tham chiếu từ người dùng (nhấn chuột vào một ô khi soạn thảo hộp này chèn tham chiếu vào ô đó) |
| 11 | Nhóm nút tuỳ chọn, được sử dụng để định nghĩa một nhóm nút tùy chọn (mọi nút tùy chọn theo ngay sau khoản mục này đều ở trong cùng nhóm nút tùy chọn) |
| 12 | Nút tùy chọn |

Bảng 4.1. Các mã kiểu khoản mục cho bảng định nghĩa hội thoại
| Mã | Мџс |
|----|--|
| 13 | Hộp kiểm tra |
| 14 | Khung, được sử dụng để vẽ khung trong hộp thoại |
| 15 | Hộp danh sách, được sử dụng để hiển thị một danh sách các khoản mục để lựa chọn (đặt tham chiếu vào danh sách trong cột văn bản của bảng định nghĩa hội thoại) |
| 16 | Hộp danh sách kết nối, khi đã nói trước cùng với hộp soạn thảo văn bản, khoản mục được lựa chọn trong danh sách được đặt trong hộp soạn thảo văn bản |
| 17 | Biểu tượng, đặt một số 1 (lựa chọn), 2 (thông tin), hoặc 3 (sự báo nguy) trong cột văn bản để chọn biểu tượng |
| 18 | Hộp danh sách tệp tin nối (chỉ có trong Windows), khi được nói trước một hộp soạn thảo văn bản và được theo sau bởi một ổ đĩa nối và hộp danh sách thư mục, kiểu mẫu trong hộp soạn thảo văn bản được sử dụng để chọn các tệp đã liệt kê |
| 19 | ổ đĩa nối và hộp danh sách thư mục (chỉ trong Windows) khi được nói trước với hộp danh sách tệp tin nối và được theo sau bởi một nhãn văn bản, nhãn văn bản này sẽ chứa ổ đĩa |
| 20 | Văn bản thư mục (chỉ trong Windows), được sử dụng để hiển thị duy chỉ thư mục hiện thời |
| 21 | Hộp danh sách thả xuống, được sử dụng để làm cho danh sách thả xuống khi được lựa chọn (đặt tham chiếu tới danh sách trong cột văn bản của bảng định nghĩa hội thoại) |
| 22 | Hộp danh sách tổ hợp thả xuống, khi được nói trước một hộp soạn thảo văn bản, mục lựa chọn trong danh sách được chèn vào trong hộp soạn thảo văn bản |
| 23 | Nút Picture, một nút được tạo ra có hình ảnh được vẽ bằng công cụ vẽ |
| 24 | Nút Help, được sử dụng để hiển thị tiêu đề Help cho hộp thoại (tên của tệp Help đi vào ô trên cùng bên trái của bảng định nghĩa hộp thoại) |

4.5. KHÁI NIỆM VỀ VISUAL BASIC FOR APPLICATIONS

Vào năm 1994, hãng phần mềm khổng lồ Microsoft đã tāng cường đáng kể khả năng Macro với mong muốn làm sao để Visual Basic trở thành ngôn ngữ Macro cho tất cả các ứng dụng tiêu chuẩn của nó, và Excel đã trở thành một ứng dụng đầu tiên trong số những ứng dụng đã được thay đổi này.

Visual Basic có tác dụng như một ngôn ngữ để phát triển các ứng dụng Windows nhờ ưu điểm của giao diện nhìn thấy của nó (sẽ tự tạo ra ứng dụng của riêng bạn) và nhờ sự đơn giản tương đối của ngôn ngữ Basic hiện đại của nó. Tuy nhiên Visual Basic là một ngôn ngữ lập trình hiện đại, bao gồm việc tạo kết cấu khối, các thủ tục, và đối tượng lập trình. Excel có một biến thể của Visual Basic có tên là Applications Edition trong Visual Basic hay Visual Basic for Applications. Nó không phải là ngôn ngữ Visual Basic đầy đủ, nhưng nó có hầu hết các thủ tục. Ngoài ra, nó có các đối tượng lập trình đặc biệt đứng ra sau nó để truy nhập các ô trên bảng tính và điều khiển Excel. Nếu bạn đã có ít nhiều kinh nghiệm lập trình bằng ngôn ngữ Pascal, C hoặc Fortran, bạn sẽ rất thoải mái với ngôn ngữ Visual Basic A. Mặt khác, nếu bạn chỉ quen lập trình bằng ngôn ngữ GW- Basic hoặc Basic, bạn cũng sẽ chấp nhận ngôn ngữ này nhưng sẽ phải học thêm khá nhiều bởi vì ngôn ngữ Basic hiện đại khác nhiều so với các phiên bản này trước kia.

Ngôn ngữ Visual Basic for Applications là một ngôn ngữ lập trình hoàn toàn theo chức năng và xứng đáng dành một cuốn sách riêng biệt để mô tả tất cả các khả năng của nó. Do khuôn khổ hạn chế của sách này nên sau đây sẽ chỉ mô tả khái quát về các khả năng mà các nhà khoa học hoặc kỹ sư có thể tận dụng. Nếu bạn đọc cần biết nhiều hơn về ngôn ngữ này, xin hãy xem thêm các cuốn sách khác như đã liệt kê trong phần "Danh mục sách tham khảo" ở cuối sách này.

4.6. TẠO CÁC THỦ TỤC VISUAL BASIC

Một thủ tục Visual Basic for Applications là một khối mã Visual Basic được thực hiện như một đơn vị. Nó được xác định theo tiêu đề mở đầu và tiêu đề kết thúc của thủ tục bao quanh khối mã cần thực hiện. Tiêu đề mở đầu của thủ tục có cú pháp như sau:

Sub ProcName (arguments)

Ở đây *ProcName* được thế chỗ bằng tên thủ tục và *arguments* là bất kỳ đối số nào được truyền tới thủ tục hoặc truyền từ thủ tục. Tiêu đề kết thúc của thủ tục rất đơn giản:

End Sub

mà đứng tiếp ngay sau câu lệnh cuối cùng của thủ tục.

Các thủ tục được đặt trên một trang modul riêng biệt trong bảng tính Excel. Bạn thêm một trang modul vào một bảng tính Excel bằng cách sử dụng câu lệnh *Insert > Macro > Modul*. Khi bạn chọn câu lệnh đó, một trang modul Excel mới được chèn vào bảng tính hiện thời, và thanh công cụ của Visual Basic được hiển thị. Thanh công cụ của Visual Basic chứa các câu lệnh để chạy, dừng và tạm dừng chương trình Visual Basic. Hình 4.3 biểu diễn thanh công cụ Visual Basic và liệt kê các hàm của các nút. Ngoài thanh công cụ, bảng chọn *Run* và *Tools* còn chứa các câu lệnh để chạy và gỡ rối các thủ tục.



Hình 4.3: Thanh công cụ Visual Basic chứa các nút để chạy và gỡ rối một thủ tục (trong Excel 5).

Ngoài các trang modul, có các bảng thoại để tạo các hộp thoại hoàn thiện với các nút, nhãn, các hộp kiểm tra, các danh sách và các hộp soạn thảo. Bạn có thể thêm một bảng thoại mới bằng cách chọn câu lệnh *Insert > Macro > Dialog*. Bảng thoại chứa một hộp thoại trống chỉ có các nút OK và Cancel. Bên cạnh hộp thoại là thanh công cụ Forms có các nút để tạo các nút, các hộp soạn thảo, nhãn, các danh sách và các đối tượng khác trên biểu mẫu. Để tạo một đối tượng, bạn chỉ cần nhấn vào một nút trên thanh công cụ và kéo đối tượng trên hộp thoại.

Một khi tạo một đối tượng trên một hộp thoại, bạn phải gán một thủ tục Visual Basic cho đối tượng đó. Hãy lựa chọn một đối tượng trên hộp thoại, chẳng hạn như một nút, và chọn câu lệnh *Tools > Assign Macro*. Hộp thoại *Assign Macro* sẽ hiển thị một danh sách tất cả các thủ tục hiện có trong trang tính này, qua đó có thể lựa chọn một thủ tục để gán vào đối tượng của bạn. Từ đây trở đi, khi bạn nhấn vào đối tượng này, thủ tục Visual Basic sẽ được thực hiện.

* Khái niệm về Lập trình hướng đối tượng:

Lập trình hướng đối tượng là một kiểu lập trình mà trong đó dữ liệu (các đặc tính) và mã để thao tác trên dữ liệu đó (các phương pháp) kết hợp với nhau trong một một đối tượng. Ví dụ như trong Visual Basic, một nút là một đối tượng, nó chứa dữ liệu mô tả vị trí của nó trên hộp thoại, nó rộng ra sao và màu sắc như thế nào và văn bản này được hiển thị ở phần dầu. Nó cũng chứa mã để kéo nút trên màn hình, để mô phỏng một nút đang được nhấn và khởi động một thủ tục xác định người dùng khi bạn nhấn nó.

Để sử dụng một đối tượng, bạn phải hoặc thay đổi các *Properties* (các đặc tính) của nó, hoặc bạn thực hiện mã gắn vào nó bằng cách gọi các *Methods* (các phương pháp) của nó. Chẳng hạn, đối tượng *Button* có các đặc tính *Top* và *Left* để đặt vị trí của nó tương ứng với góc trái trên đỉnh của hộp thoại. Một chương trình có thể đọc các đặc tính này để biết được vị trí một nút, hoặc thay đổi chúng để chuyển một nút tới một vị trí khác. Một nút cũng có phương pháp *Copy* (nếu đã thực hiện) mà đặt bản sao của đối tượng nút trên bảng nhớ tạm.

Các đối tượng được kết hợp vào các đối tượng mới bằng cách sử đụng mẫu lớp chứa, mỗi trường hợp của một đối tượng được chứa trong một đối tượng lớn hơn, và đối tượng lớn hơn này lại chứa trong một đối tượng thậm chí còn lớn hơn. Bởi vậy, các nút trên một hộp thoại được chứa trong đối tượng của hộp thoại. Hộp này được chứa trong đối tượng của trang tính cùng với các đối tượng bảng tính và đối tượng modul. Tất cả các đối tượng của trang tính mở được chứa trong đối tượng ứng dụng và cứ như vậy. Khi bạn truy nhập một đối tượng, bạn sử dụng trật tự lớp chứa này để xác định một cách chính xác đối tượng nào mà bạn đang truy nhập.

4.7. TRUY NHẬP CÁC ĐỐI TƯỢNG BẰNG VISUAL BASIC

Thực ra Visual Basic for Applications không hoàn toàn là một ngôn ngữ hướng đối tượng, trong đó bạn không thể tạo các đối tượng mới, nhưng nó chứa và sử dụng các đối tượng khi truy nhập Excel. Mỗi mục tin trong Excel là

một đối tượng, và có thể truy nhập được nhờ một chương trình Visual Basic. Không chỉ là các ô và các đối tượng bảng tính, mà còn là các câu lệnh và các hàm của bảng tính.

Trong Visual Basic for Applications, một ô hoặc một nhóm các ô là đối tượng Range được chứa trong đối tượng bảng tính. Việc truy nhập một đặc tính của đối tượng đứng sau trật tự lớp chứa này sẽ xác định đối tượng và đặc tính mà bạn quan tâm. Chẳng hạn, để truy nhập giá trị của ô *A1* trên bảng tính có tên *FlowCalc* trong trang tính *Design*, bạn sẽ sử dụng cú pháp sau đây:

APPLICATIONS. WORKBOOK ("*Design*"). SHEETS ("*FlowCalc*"). RANGE ("*A1*"). VALUE.

Cú pháp này đặt sau mỗi đối tượng lớp chứa một dấu chấm, rồi đến đối tượng bị chứa trong nó. Dấu cảm thán (!) cũng được sử dụng để tách các đối tượng hoặc đặc tính có cùng tên. Nếu sử dụng dấu chấm, có nghĩa là đặc tính sẽ được lựa chọn, nếu sử dụng dấu cảm thán thì có nghĩa là đối tượng sẽ được lựa chọn.

Bạn không phải luôn gõ tất cả các nổi kết đã thấy ở trên. Nếu bạn không sử dụng các nối kết trên cạnh trái của trật tự các lớp chứa, thì các đối tượng hoạt động hiện thời sẽ được thừa nhận là có ở đó. Bởi vậy, thông thường bạn chỉ cần xác định Sheet và Range cho một ô hoặc phạm vi các ô. Chẳng hạn, đối tượng sau sẽ làm việc trong hầu hết các trường hợp:

SHEETS ("FlowCalc"). RANGE ("A1")

Ngoài ra, có một số đối tượng đặc biệt luôn luôn lựa chọn đối tượng hoạt động tạm thời. ActiveSheet sẽ lựa chọn bảng hoạt động, ActiveCell lựa chọn ô hoạt động và Selection trở về các đối tượng lựa chọn trên đối tượng lớp chứa lựa chọn. Chẳng hạn, ActiveSheet. Selection truy nhập ô hoạt động trong lựa chọn đó, và ActiveDialog. Selection truy nhập điều khiển lựa chọn hiện thời trên hộp thoại hoạt động.

Chú ý: Để biết một đối tượng có Properties và Methods nào, hãy sử dụng Object Browser (nhấn nút Object Browser trên thanh công cụ Visual Basic). Trong hộp thoại Object Browser, lựa chọn thư viện (Excel hoặc VBA) và đối tượng. Một danh sách các Properties và Methods hiện có sẽ xuất hiện. Việc lựa chọn một Property và Method đặc biệt sẽ hiển thị cú pháp của nó và việc nhấn nút hỏi chấm (?) sẽ hiển thị chủ để trợ giúp mà mô tả nó.

4.8. TẠO CÁC THỦ TỤC CÂU LỆNH ĐỂ TỰ ĐỘNG HOÁ CÁC BẢNG TÍNH

Các thủ tục cơ bản nhất trong Visual Basic là dùng để tự động lặp lại các hoạt động trên bảng tính. Các thủ tục kiểu này được biết đến như những thủ tục câu lệnh, bởi vì chúng được thực hiện giống như những câu lệnh bảng chọn. Hầu như mọi nhiệm vụ lặp lại đều có thể được thực hiện nhờ một thủ tục. Tác dụng chính cho các thủ tục này là lưu giữ cho bạn thời điểm khi bạn phải thực hiện những hoạt động lặp lại và tạo các bảng tính đã tự động hoá cho những người không phải là chuyên gia trong Excel. Chúng cũng

rất có thể làm cho nhiều người phải ngạc nhiên khâm phục, bởi vì các bảng tính dường như sẽ tạo ra chính chúng mà không cần chúng ta phải chạm tay vào bàn phím.

4.8.1. Ghi một thủ tục

Việc tạo các thủ tục câu lệnh là đơn giản và việc sử dụng chúng thậm chí còn đơn giản hơn. Để tạo một thủ tục câu lệnh, hãy thiết lập một bảng tính như bảng tính trước khi thủ tục được thực hiện, và chọn câu lệnh *Tools > Record Macro > Record New Macro*. Hộp thoại *Record New Macro* xuất hiện - nhấn nút lựa chọn (*Options*), và hộp thoại sẽ mở rộng như biểu diễn ở đây.

Trong hộp thoại này, bạn gõ một tên cho một thủ tục của bạn và một cách tuỳ chọn, đặt nó trên bảng chọn *Tools* và gán nó một phím cất sử dụng trong tổ hợp với phím Ctrl như một phím cất để mở thủ tục. Bạn không cần xác định tên bảng chọn hoặc phím cất - bạn có thể gán chúng sau đó, hoặc gán thủ tục vào một nút. Hộp thoại *Record New Macro* cũng bao gồm sự lựa chọn giữa việc ghi thủ tục trong trang tính hiện thời, một trang tính dành riêng hoặc một trang tính mới. Đặt các thủ tục mà bạn dự định sử dụng trên nhiều trang tính khác nhau trong trang tính dành riêng, bởi vì trang tính đó được mở bất cứ khi nào bạn chạy Excel. Đặt các thủ tục riêng của dự án trong trang tính tương tự như dự án.

| Record Macro | | <u>?</u> × |
|------------------------|----------------------------------|------------|
| Macro name: Macro1 | | |
| Shortzut key: Ctrl+ | Store macro in: This Workbook | • |
| Description: | | |
| Macro recorded 3/1 | 1 2/2004 by bm ОК | Cancel |

Hình 4.4. Hộp thoại Record Macro.

Phương án chọn ngôn ngữ phải là Visual Basic. Excel có khả năng tạo các Macro trong ngôn ngữ Excel Macro của các phiên bản đời trước đó, nhưng bạn chỉ nên sử dụng ngôn ngữ đó nếu có một lý do cụ thể nào đó để thực hiện như vậy, chẳng hạn như việc tạo một thủ tục cho những người dùng không có thêm các phiên bản Excel gần đây.

Ngay khi bạn nhấn OK, thanh công cụ *Stop* dường như chứa nút *Stop Marco*, va Excel bắt đầu ghi các tác dụng của bạn. Tất cả những gì ban thực hiện đều được ghi trên bảng Macro: việc mở và đóng tệp, gõ các công thức, van shi bạn được kết thúc, hãy chọn câu lệnh *Tools* > *Record Macro* > *Stop Recording* hoặc nhấn nút *Stop Macro*.

Nếu bạn đã ghi thủ tục của mình trên một trang modul, hãy chuyển sang bảng đó bằng cách sử dụng các Tab của bảng tại đáy của sổ bảng tính, và bạn sẽ thấy thủ tục mới đó. Nếu thủ tục của bạn nằm trên trang tính dành riêng, hãy sử dụng câu lệnh Unhide trên bảng chọn Windows để làm cho trang tính dành riêng trở thành trang tính nhìn thấy.

4.8.2. Thêm vào một lệnh ghi hiện thời

Để ghi một thủ tục thứ hai trên cùng một trang modul, hoặc để thêm vào một thủ tục hiện thời, hãy lựa chọn vị trí nơi bạn muốn bắt đầu ghi và chọn câu lệnh Tools > Record Marco > Record at Mark để bắt đầu ghi. Khi bạn kết thúc ghi, hãy sử dụng Tools > Record Macro > Stop recording hoặc nút Stop Macro như trước kia.

Câu lệnh *Tools > Record Macro > Record at Mark* không chèn tiêu đề mở đầu của thủ tục cho mã đã ghi và không gọi tên cho chúng. Nó chỉ chèn mã đã ghi tại vị trí đánh dấu. Trừ phi mã này được chèn trong một thủ tục hiện thời, bạn phải gọi tên cho mã, từ đó bạn có thể truy nhập nó. Để gọi tên cho mã đã ghi, hãy chèn tiêu đề trên và dưới của thủ tục phía trên và phía dưới mã đã ghi. Tên được dùng trong tiêu để mở đầu của thủ tục là tên dùng để truy nhập mã.

Chú ý: Khi bạn đang ghi các Macro, tham trỏ ô được ghi trong chế độ tuyệt đối (Absolute mode), nó giúp cho mỗi lần có thể truy nhập các ô giống nhau. Chẳng hạn, nếu tạo một Macro định dạng ô thì bạn sẽ muốn Macro định dạng ô hoạt động hiện thời, chứ không phải ô hoạt động khi bạn đã ghi Macro. Để thay đổi việc ghi trong chế độ tương đối (Relative mode), chọn Tools > Record Macro > Use Relative References. Hãy chọn lại câu lệnh này để trở lại như trước.

4.8.3. Gán một thủ tục cho một đối tượng

Thay vì cho một số thủ tục một phím cắt hoặc thay vì thực hiện nó từ bảng chọn Tools, bạn có thể gán một thủ tục cho một đối tượng, chẳng hạn như một nút và thực hiện nó bằng cách nhấn vào đối tượng này. Bạn có thể tạo một nút bằng công cụ *Create Button* trên thanh công cụ *Forms* hoặc *Drawing* nhưng bất cứ đối tượng nào trên một bảng tính cũng đều có thể buộc một thủ tục phải gán cho nó, kể cả một biểu đồ nhúng hoặc một phần tử vẽ (chẳng hạn như một hình chữ nhật hoặc một đường thẳng). Để gán một thủ tục cho một đối tượng, lựa chọn đối tượng, chọn câu lệnh *Tools* > *Assign Macro*, chọn thủ tục để gán từ hộp thoại *Assign Macro* và nhấn OK.

Để thay đổi thủ tục đã gán cho một nút, trước tiên bạn cần chọn nút đó, chứ không lựa chọn nút thực hiện thủ tục. Để lựa chọn một nút mà không cần thực hiện thủ tục đã gán này, hãy giữ phím Ctrl xuống khi lựa chọn nó. Ngoài ra, bạn có thể sử dụng công cụ *Drawing Selection* trên thanh công cụ *Drawing* để chọn một đối tượng mà không cần thực hiện thủ tục đã gán. Phương pháp này dùng cho tất cả các đối tượng có những thủ tục đã gán cho chúng.

4.8.4. Chạy một thủ tục câu lệnh

Để chạy một thủ tục câu lệnh, giữ phím Ctrl xuống và nhấn phím cắt của nó (nếu bạn đã gán một thủ tục), nhấn vào đối tượng của nó (nếu bạn đã gán một thủ tục cho một đối tượng), hoặc lựa chọn *Tools > Macro* và chọn tên của nó từ hộp thoại Macro.

Sau đây nêu ra một ví dụ, hãy tạo một thư mục câu lệnh để định dạng và tạo dáng các ô. Đây là một thư mục đơn giản mà định dạng những nội dung của ô lựa chọn hoặc ô là Scientific với hai vị trí thập phân và vẽ một hộp xung quanh các ô này. Gán thủ tục cho một nút trên bảng tính.

1. Hãy bắt đầu với một bảng tính mới, và gọi tên nó như trong hình 4.4.

2. Lựa chọn ô B7 (trên thực tế thì ô nào cũng được) và nhập một con số nào đó vào nó để sử dụng trong khi ghi thủ tục.

3. Chọn câu lệnh Tools > Record Macro > Record New Macro.

4. Trong hộp thoại Record New Macro, nhấn nút Options.

5. Gọi tên *Macro* là *Formatlt*, gõ một mô tả, đặt nó trên bảng chọn *Tools*, thiết đặt phím cắt là F, và nhấn OK để bắt đầu ghi.

6. Chọn câu lệnh *Tools > Record Macro > Use Relative References* (nó đặt một dấu kiểm tra cạnh câu lệnh). Nếu nó đã được kiểm tra, để nó lại một mình vì bộ ghi đã ở trong chế độ Relative.

7. Nhấn vào ô có số, chọn câu lệnh *Format* > *Cells*, tab *Number*, lựa chọn đình dạng Scientific với hai vị trí thập phân và nhấn OK.

8. Nhấn mũi tên đi xuống trên nút Borders và chọn nút vẽ từ bảng màu.

9. Nhấn vào nút Stop Macro.

10. Chọn Tab của trang modul 1.

Macro xuất hiện trên bảng Macro. Bộ ghi tự động chèn một số ghi chú ở phần đỉnh cùng với tiêu đề mở đầu và tiêu đề kết thúc của thủ tục và tự động ghi tất cả các hoạt động thực hiện trên ô lựa chọn.

Chú ý: Các ghi chú trong Visual Basic là một văn bản nào đó trên một dòng mà có một dấu đóng ngoặc đơn đằng sau. Các ghi chú có thể xuất hiện trên một dòng đơn, hoặc có thể xuất hiện ở bên phải của câu lệnh có thể thực hiện được. Tự do sử dụng các ghi chú trong các mã của bạn để làm cho chúng có ý nghĩa hơn.

Câu lệnh đầu tiên trong thủ tục đứng sau tiêu đề mở đầu của thủ tục là một câu lệnh lựa chọn. Chính câu lệnh là kết quả của việc thiết đặt việc ghi Relative và việc lựa chọn ô B7.

ACTIVECELL.OFFSET (6, 1). RANGE ("A1"). SELECI'.

Đối tượng ActiveCell quyết định ô lựa chọn hiện thời trong bảng chọn trên cùng. Trong trường hợp này, nó là ô A1. Phương pháp *Offset* (6,1) di chuyển ô ban đầu cho một lựa chọn xuống 6 dòng và sang phải 1 cột (ô B7). Phương pháp *Range* ("A1") trở về ô trái - trên cùng trong phạm vi dịch chuyển mà nó được dùng. Trong trường hợp này, lệnh *Offset* đã làm cho ô B7 trở thành ô trái - trên cùng, do vậy việc lựa chọn ô trái - trên cùng (A1) trong phạm vi này sẽ lựa chọn ô B7. Nếu lựa chọn nhiều hơn một ô, phương pháp *Range* sẽ có một phạm vi các ô trong đối số. Phương pháp *Select* lựa chọn ô đó, làm cho nó trở thành ô hoạt động mới.

Sau câu lệnh Select là năm dòng được câu lệnh Borders thiết đặt. Khi bạn chọn một đường viền bằng cách sử dụng hộp thoại Border hoặc nút Borders, bạn thiết đặt một hoặc nhiều hơn một trong số năm lựa chọn: Left, Right, Bottom, hoặc Outline. Mỗi trong số năm lựa chọn này thu được một dòng trong thư mục đã ghi, dẫu cho bạn không thay đổi lựa chọn từ việc thiết đặt ngầm định. Trong ví dụ này, viền ngoài được lựa chọn, do vậy tất cả các lựa chọn được thiết đặt là không, ngoại trừ lựa chọn BorderAround, nó được thiết đặt là đường đậm vừa.

Chú ý:

Bảy biến trong hình 4.4 bắt đầu bằng xl (xlLeft, xlRight, xlTop, xlBottom, xlNone, xlAutomatic, và xlMedium) là các hằng xác định toàn bộ cho sử dụng với các ứng dụng Visual Basic. Các hằng hiện có được liệt kê trong trợ giúp trực tiếp, và cũng được liệt kê trong Object Browser thuộc đối tượng Constant trong Excel và các thư viện Visual Basic. Bạn có thể sử dụng những hằng này ở bất cứ nơi nào giá trị được sử dụng để làm cho mã của bạn dễ đọc hơn.

Thủ tục này được dự định sẽ định dạng bất cứ ô nào hoặc các ô đã lựa chọn (không cần thiết ô mà *xuống dưới 6 hàng* và sang phải một cột tính từ ô hoạt động), do vậy bạn phải dời chuyển câu lệnh lựa chọn. Tiếp theo, tạo một nút trên bảng chọn và gán thủ tục cho nó.

 Xoá dòng đầu tiên của thủ tục, hoặc đổi nó thành cột ghi chú bằng cách gõ một dấu ngoặc đơn ở phía xa, bên trái.

2. Chọn Tab của bảng cho bảng tính hình 4.4.

3. Nhấn vào nút Drawing trên thanh công cụ Standard để hiển thị thanh công cụ Drawing.

4. Nhấn vào nút công cụ *Create* và kéo một nút trên bảng tính bằng cách nhấn và kéo trong một vị trí thích hợp.

5. Trong hộp thoại Assign Macro xuất hiện, lựa chọn Macro Formatlt và nhấn OK.

6. Sử dụng trỏ chuột để lựa chọn văn bản trên bề mặt của nút, sau đó gõ Formatlt. Nếu bạn không lựa chọn nút này nữa vì một lý đo nào đó, hãy giữ phím Ctrl xuống khi lựa chọn nó lần nữa, do vậy bạn không thực hiện thủ tục gán này. Thủ tục này hiện thời đã sắn sàng để sử dụng. Gõ một số chữ số trong một số ô, sau đó lựa chọn các ô và nhấn vào nút *Formatlt* để biết được những gì xáy ra. Lựa chọn một số ô khác và chọn câu lệnh *Tools* > *Formatlt*. Thủ tục tương tự được thực hiện. Lựa chọn tập hợp các ô thứ ba và nhấn Ctrl +F và thủ tục lại được thực hiện.

Một khi đã tạo, một trang mođul chỉ được lưu giữ hoặc mở giống như một bằng tính. Để sử dụng một thủ tục, trang modul chứa nó phải là bằng mở. Các thủ tục bạn tạo đều là đa năng; bạn có thể sử dụng các thủ tục trên một trang modul mở có một bằng tính mở nào đó, không chỉ những thủ tục mà hoạt động khi chúng được tạo.

4.9. LẬP TRÌNH VỚI VISUAL BASIC FOR APPLICATIONS

Các thủ tục sẽ không dơn giản như thủ tục trong ví dụ trước. Bạn có thể ghi các chương trình hoàn thiện bao gồm các hộp thoại khách hàng và các bảng chọn, các phép tính lặp lại, sự truy nhập và tạo tệp, và điều khiến khách hàng của một bảng tính bằng cách sử dụng Visual Basic. Ngoài ra, nếu bạn có một hàm khách hàng mà cần chạy nhanh với tốc độ của ngôn ngữ biến dịch, thì hãy tạo nó trong một modul trong Dynamic Link Library (DLL) và sau đó sử dụng câu lệnh *Declare* dễ ghi nó vào sổ bằng Excel. Một thủ tục bên ngoài nào đó có thể được sử dụng rất giống với thủ tục gắn vào một khi nó được ghi sổ bằng Excel.

4.9.1. Thao tác các tham tró ô

Khi bạn sử dụng Visual Basic để thao tác trên Excel, nói chung bạn sẽ truy nhập các ô của bảng tính và các vùng ô. Một vùng ô nào đó trên bảng tính (từ một ô đơn lẻ tới một vùng rộng) là một đối tượng *Range*. Một vùng ô được truy nhập bằng cách sử dụng phương pháp Range, với các ô sẽ truy nhập được liệt kê trong đối số của phương pháp Range. Ví dụ: *Range ("B7: F9")* lựa chọn các ô trong vùng ô B7:F9 hình chữ nhật. Nếu bạn áp dụng phương pháp phạm vi khác cho vùng hiện thời thì bộ chọn sẽ xem xết phạm vi khi chúng nhận nó và nói chung không áp dụng cho bảng tính. Chẳng hạn:

```
ACTIVESHEET.RANGE ("B7:F9"). RANGE ("B2")
```

sẽ truy nhập ô C9, dù cho tham trỏ đã xác định trong phương pháp vùng trước là B2. Phương pháp Range đầu tiên trở lại phạm vi B7:F9, phương pháp Range thứ hai đưa ra ô mà xuống dưới hai hàng và thấp hơn so với góc phải trên đỉnh của vùng được ấp dụng hai hàng.

Phương pháp vùng hữu dụng khác là *Cells* (). Phương pháp Cells sử dụng hai đôi số nguyên cho hàng và cột đã chỉ báo. Công dụng lớn nhất của Cells là trong một vòng lặp hoặc cấu trúc khác, nơi mà bạn cấn truy nhập các ô trong vùng bằng cách sử dụng các chỉ số nguyên. Chẳng hạn:

ACTIVESHEET. RANGE (Cells (7,2), Cells (9,6))

là tương tự như:

ACTIVESHEET. RANGE ("B7:F9")

4.9.2. Các biến Visual Basic

Các biến trong Visual Basic không cần được khai báo nhưng được tự động tạo ra lần đầu tiên khi bạn sử dụng chúng. Tuy nhiên, dẫu sao bạn cũng phải khai báo chúng nhằm giúp xác dịnh vị trí những lồi sai chính tả và các vấn để khác. Bạn có thể tự bắt mình khai báo tạt cả mọi thứ bằng cách sử dụng câu lệnh *Option Explicit* tại đỉnh tất cả các modul của bạn. Nếu một biến không được xác định thì một lỗi thời gian biên dịch sẽ là do Visual Basic. Lỗi này thường gây ra do viết sai chính tả một tên biến hiện thời.

Các biến được khai báo bằng cách sử dụng các câu lệnh *Dim*, *Internal*, *External*, *Const* và *Static*. Câu lệnh *Dim* khai báo một biến cho thủ tục hiện thời nếu nó được sử dụng trong thủ tục đó, hoặc cho tát cả các thủ tục trong một modul nếu nó được khai báo tai đính của modul, bên ngoài thủ tục nào đó. Cú pháp của câu lệnh *Dim* là:

Dim Variable As Type

Từ *Variable* được thay thể bằng tên của biến mà bạn đang khai báo và *Type* được thay thế bằng khai báo *Type* để áp dụng cho biến.

Thủ thuật: Có một số quy ước để định tên các biến. Một trong số những biến đễ đọc nhất mà tạo tiền tố tên biến theo kiểu của nó sẽ giúp cho việc nhớ kiểu biến trở nên đễ dàng. Chẳng han, **intCounter** cho một bộ đếm nguyên hoặc **StrFileName** cho một chuỗi.

Có 10 kiểu dữ liệu được tạo trong Visual Basic. Danh sách sau đây sẽ biểu diễn các kiểu và các tiền tổ được xem xét cho các biến kiểu đó.

| Kiểu | Tiến tố | Mó tả | | |
|----------|-------------------------|---|--|--|
| Integer | int | Giá trị nguyên 16 bit | | |
| Long | Ing | Giá trị nguyên 32 bit | | |
| Single | sng | Dấu chấm di động chính xác đơn (32 bit) | | |
| Double | dbl | Dấu chấm di động chính xác kép (64 bit) | | |
| Currency | cur | Kiểu nguyên 64 bit để tính toán tiền tệ | | |
| Date | Jate | Một ngày hoặc một khoảng thời gian nào đó | | |
| String | str | Các chuỗi văn bản | | |
| Variant | var | Kiểu dữ liệu nào đó, kiểu thực được xác định vào thời gian chay chượng trình | | |
| Boolean | ſ | True logic hay False logic | | |
| Object | phu thuộc vào đối tượng | Một biến đối tượng theo kiểu nào đó | | |

Ngoài ra, có một kiểu biến *Array*, nó là một nhóm các biến có cùng tên và được truy nhập bằng cách sử dụng một chỉ số số nguyên.

Chú ý:

Kiểu dữ liệu Variant là kiểu đặc biệt, trong đó nó có thể được gán giá trị của một kiểu nào đó. Kiểu bề ngoài của một Variant được gán tại thời gian chạy chương trình. Các kiểu Variant cũng lưu giữ các kiểu dữ liệu đặc biệt chẳng hạn như các bảng cơ sở dữ liệu. Trở ngại khi sử dụng các Variant đó là kiểu của chúng không được kiểm tra vào thời gian biên dịch, nhưng lại được kiểm tra vào thời gian chạy chương trình ở nơi chúng tạo ra các lỗi thời gian chạy chương trình. Điều này cũng liên quan nhiều đến việc hiểu kiểu của Variant ra sao và liên quan đến việc lưu giữ nó.

Câu lệnh *Internal* làm việc y hệt như câu lệnh *Dim*, nhưng lại quan trọng hơn, bởi vì nó diễn đạt tốt hơn phạm vi thực của các biến được khai báo bởi câu lệnh. Tuy nhiên, nó được sử dụng ngẫu nhiên vì *Dim* dùng phổ biến trong nhiều năm.

Câu lệnh *External* có cú pháp tương tự như câu lệnh *Dim*, nhưng phạm vi của nó rộng hơn nhiều. Câu lệnh *External* được sử dụng để khai báo một biến tại đỉnh của một modul. Mặc dù vậy, trong trường hợp này, nó không chỉ có sẵn bên trong modul đó, nó còn có sẵn đối với tất cả các modul trong một ứng dụng (có nghĩa nó là một biến chung).

Cần lưu ý trong khi các biến chung là hữu dụng vì chúng có ở mọi nơi, hãy cẩn trọng vì chúng cũng có thể bị thay đổi ở một nơi nào đó. Nếu bạn sử dụng lại một biến chung trong thủ tục khác mà không khai báo nó ở đây, thì bạn sẽ thay đổi giá trị của biến chung.

Câu lệnh Const không thực sự khai báo một biến, nhưng lại khai báo một giá trị hằng. Cú pháp của câu lệnh Const là:

CONST Variable = Value

Khi hằng số được khai báo trong một chương trình, bất cứ nơi nào tên Variable được tìm thấy trong chương trình, nó sẽ được thay thế bằng Value. Vì các hằng số được chèn là các giá trị đúng tại thời gian biên dịch, nên chúng không thể được thay đổi trong khi chương trình đang chạy. Các hằng số ban đầu được sử dụng để thiết đặt các cờ hiệu, và để làm cho các câu lệnh Visual Basic trở nên có ý nghĩa hơn. Bằng cách khai báo và sử dụng các hằng số, bạn giúp cho mã của mình đọc dễ dàng hơn là lập mã cho các hằng đó các giá trị đúng.

Kiểu Static được sử dụng giống như Dim trong thư mục, nhưng lại phân bố bộ nhớ tĩnh cho một biến. Cú pháp cho câu lệnh Static như sau:

STATIC Variable As Type

Thông thường, bộ nhớ cho các biến được phân bố khi thủ tục bắt đầu chạy và biến mất khi thủ tục hoàn thành. Lệnh khai báo *Static* phân bố bộ nhớ tại thời gian biên dịch,

do vậy nội dung của các biến *Static* không biến mất khi một thủ tục hoàn thành, nhưng lại có sẵn các lệnh gọi tương lai đối với thủ tục đó.

Thủ thuật: Việc tự chúng nhóm tất cả các khai báo chung và các hằng vào một modul đơn là rất phổ biến. Theo cách này, chúng dễ dàng phân bố và thay đổi nếu cần thiết.

4.9.3. Tạo các câu lệnh toán học bằng Visual Basic

Các lệnh gán toán học (những lệnh mà tính toán một giá trị và lưu giữ nó trong một biến) là hoàn toàn tương tự như các đương lượng đại số. Phía bên trái là một biến dùng để nhận giá trị tính toán, đứng trước là một dấu bằng và công thức để tính toán giá trị ở bên phải. Ngoài ra, phép trừ, phép nhân và phép chia là tương tự. Phép luỹ thừa được thực hiện nhờ toán tử hướng lên (^).

Các hàm toán học có tên và chức năng tương tự như các hàm bảng tính. Trên thực tế, các hàm bảng tính có thể được sử dụng trong các câu lệnh Visual Basic.

Chú ý:

Trình tự ưu tiên thực hiện của các toán tử trong Visual Basic for Applications hơi khác so với của các công thức trên bảng tính. Trình tự thực hiện của các toán tử xác định toán tử nào được thực hiện đầu tiên. Trong Visual Basic for Applications, phép luỹ thừa thực hiện trước sự phủ định, trong khi đó trên bảng tính, sự phủ định thực hiện trước tiên. Chẳng hạn, -1² trở về 1 trên bảng tính và -1 từ Visual Basic for Applications. Để có được kết quả tương tự trong Visual Basic for Applicatios như cho bảng tính. hãy sử dụng các tham số để ấn định trật tự của phép tính: (-1)².

Các kiểu của biến rất quan trọng trong Visual Basic. Nếu bạn tạo một công thức mà bao gồm các kiểu biến khác nhau, Visual Basic sẽ chuyển đổi toàn bộ các giá trị này thành kiểu chính xác nhất trước khi thực hiện các phép tính. Kết quả cuối cùng khi đó được chuyển đổi thành kiểu của biến ở phía trái công thức. Trong hầu hết các trường hợp, những chuyển đổi phải không có ảnh hưởng tới các kết quả này. Nếu nó sẽ ảnh hưởng tới các kết quả, hãy chuyển đổi tất cả các biến thành một kiểu đơn lẻ trước khi tính toán công thức. Các hàm là có sẵn để thực hiện các chuyển đổi này cho bạn

4.9.4. Xử lý đầu vào và đầu ra

Để có được một đề mục đơn của đầu vào từ người sử dụng, hãy sử dụng hàm INPUTBOX. Để dễ dàng gửi cho người dùng một thông báo, bạn hãy sử dụng hàm MSGBOX. Cả hai hàm này đều hiển thị hộp thoại chứa văn bản và các nút. Hàm MSGBOX hiển thị một thông báo cho người dùng. Hàm INPUTBOX hiển thị một thông báo và có một *Edit Box* cho người dùng để nhập thông tin vào. Ngoài những hộp thoại gắn vào, bạn có thể tạo các hộp thoại khách hàng riêng của bạn. Để tạo một hộp thoại khách hàng, hãy mở một bảng thoại và sử dụng hộp công cụ *Forms* để kéo các nút và hộp bạn cần. Để truy nhập các tệp đĩa bên ngoài, hãy sử dụng các hàm thoại chuẩn để hiển thị các hộp thoại *Save As* và *Windows Open* chuẩn cho bạn để lựa chọn hoặc nhập vào tệp bạn muốn. Các hộp thoại thoại chuẩn thực tế không mở tệp, nhưng chỉ trở về tên hợp thức và đường dẫn của một tệp. Hai hộp thoại chuẩn được sử dụng là:

answer = Application. GetSaveAsFilename (*strFileName*, *strFilter*) answer = Application. GetOpenFilename (*strFilter*)

Có nhiều đối số hơn, nhưng chỉ một số ít trong đó là tất cả những đối số cần thiết để sử dụng các hàm. Kết quả của những hàm này (được lưu giữ trong trả lời biến) là tên và đường dẫn cho tệp cần phải mở. *strFileName* chứa một tên tệp gợi ý mà được chèn vào trong hộp thoại *Save As*. Người dùng không phải sử dụng để gợi ý đó, nhưng vẫn có thể nếu họ muốn. Biến *strFilter* chứa một màn che tệp để kiểm soát những tệp được liệt kê trong hộp thoại. Nếu không có màn che nào thì tất cả các tệp được hiển thị.

Khi bạn có tên và đường dẫn, hãy sử dụng câu lệnh *Open* để chuẩn bị cho một tệp được truy nhập. Cú pháp cơ sở của câu lệnh *Open* là:

OPEN fileName For inputType As # 1

Đối số fileName lại là tên và đường dẫn của tệp. Đối số inputType lấy 4 giá trị: Input, Output, Append và Random. #1 là một số xác định tệp và nối lệnh Open với các lệnh Input và Write mà truy nhập tệp mở. Có nhiều lựa chọn hơn, có sẵn cho lệnh Open, nhưng cú pháp đơn giản này sẽ xử lý hầu hết các tình huống.

Việc ghi vào một tệp mở được thực hiện bằng câu lệnh Write và Print. Cú pháp của Write và Print là:

WRITE #1, Variable, Variable, ... PRINT #1, Variable, Variable ...

Ở đây Variable, Variable,... là danh sách của một hay nhiều biến mà các nội dung của bạn muốn ghi vào tệp và #1 là số tệp sử dụng để mở tệp. Sự khác nhau giữa Write và Print là rất lớn. Print tạo ra đầu ra mà có thể được in trên màn hình hoặc trên máy in. Các giá trị được in trong cột và có thể được định dạng, và các nội dung của các biến chuỗi được in mà không có dấu cách. Mặt khác, Write dùng để tạo ra các tệp giúp cho việc đọc lại trong chương trình Visual Basic được dễ dàng hơn. Write đặt các dấu cách quanh tất cả mọi thứ được ghi vào tệp để làm cho việc tách biệt một giá trị đã ghi này với một giá trị đã ghi khác được dễ dàng. Dấu đóng ngoặc xung quanh các chuỗi được in và các dấu phảy được đặt giữa mỗi mục in.

Để đọc thông tin từ một tệp, bạn hãy sử dụng câu lệnh *Input* và *Line Input*. *Input* và *Line Input* có cú pháp như sau:

INPUT #1, Variable, Variable,... LINE INPUT #1, StringVariable Input đọc dữ liệu từ một tệp cho đến khi tất cả các biến của nó được điền đầy. Line Input đọc một dòng của văn bản trong một biến chuỗi, lên tới tận (nhưng không kể cả) ký tự giũa dòng hồi tiếp tại cuối dòng. Câu lệnh Input bổ sung cho câu lệnh Write, và có thể tìm được các mục tin chính xác được Write ghi. Nó cũng có thể được dùng để đọc thông tin được ghi nhờ câu lệnh Print, nhưng có thể đọc các chuỗi hoàn toàn khác. Chẳng hạn, một chuỗi đon chứa dấu phảy có thể được ghi nhờ Write hoặc nhờ Print. Write sẽ đặt các dấu ngoặc kép quanh chuỗi, do vậy Input sẽ biết để gộp dấu phảy khi nó đọc lùi chuỗi trong máy tính. Print sẽ không bao quanh chuỗi bằng dấu ngoặc kép mà câu lệnh Input sẽ xem dấu phảy là một dấu cách, và sẽ ngừng nhập chuỗi tại dấu phảy.

Sử dụng *Line Input* để có một dòng văn bản hoàn chỉnh từ một tệp. *Line Input* không chú ý tới tất cả các dấu cách trong một tệp ngoại trừ ký tự dòng mới và phần cuối của tệp. Dấu phảy, dấu ngoặc kép, tất cả mọi thứ được gộp trong chuỗi sẽ đọc trong chương trình, lên tới nhưng không gồm ký tự dòng mới.

Chú ý: Trên máy tính MS DOS hoặc Windows, ký tự dòng mới là sự quay lại đầu dòng, cặp ký tự giữa dòng.

Khi bạn thực hiện với một tệp, hãy đóng nó bằng câu lệnh Close. Cú pháp là: CLOSE #1.

Ở đây, #1 lại là số tệp mà nhờ nó tệp được mở.

Nếu bạn mở một tệp trong một chương trình và chương trình này hỏng trước khi bạn thực hiện câu lệnh *Close*, tệp sẽ vẫn mở cho đến khi bạn thoát khỏi Excel. Nếu bạn lại chạy chương trình của bạn thì nó sẽ hỏng khi cố mở tệp do lỗi *File Open*. Để đóng tệp, hãy thực hiện câu lệnh *Close* trong ô *Immediate* của cửa sổ *Debug*. Bạn cũng có thể tạo một thủ tục đơn lẻ bằng câu lệnh *Close* đơn lẻ trong nó. Khi bạn chạy thủ tục, nó sẽ đóng bất cứ cái gì mở tệp.

4.9.5. Điều khiển lưu đồ của chương trình

Lưu đồ tác vụ trong một chương trình đang chạy được điều khiển nhờ các vòng lặp, bước nhảy, các điều kiện, hàm và các lệnh gọi thủ tục con. Các vòng lặp được thực hiện bằng các câu lệnh *For/Next, DoLoop* và *For Each/Next.* Câu lệnh *For/Next* tạo một vòng lặp được đếm. Cú pháp của vòng lặp *For/Next* như sau:

```
FOR loopCounter = Start TO end STOP Step
......
Block of Code
.....
NEXT loopCounter
```

Ở đây, *loopCounter* là một biến mà thay đổi cho mỗi vòng lặp qua khối mã, start là giá trị ban đầu của bộ đếm vòng lặp và end là giá trị cuối. Step là lượng để thay đổi bộ

đếm vòng lặp trong thời gian mỗi vòng lặp. Chẳng hạn, vòng lặp sau đây sẽ lặp lại khối các câu lệnh 10 lần nhờ bộ đếm vòng lặp, I, lấy các giá trị 2, 4, 6, 8, 10, 12, 14, 16, 18 và 20.

FOR *I* = 2 TO 20 Step 2

•••••

Block of Statements

•••••

NEXT I

Câu lệnh For xác định giá trị ban đầu cho toàn bộ bộ đếm vòng lặp, giá trị cuối và bước. Nếu bước là 1, mệnh đề Step1 là tuỳ chọn. Bạn sử dụng vòng lặp được đếm khi muốn thực hiện cho một khối các câu lệnh một số lần cụ thể.

Các câu lệnh *Do/Loop* thực hiện một vòng lặp đầu cuối theo lôgíc. Có nghĩa là, giá trị lôgíc xác định khi vòng lặp dừng thực hiện. Vòng lặp *While/Wend* khác là trường hợp dặc biệt của các câu lệnh *Do/Loop* có cú pháp sau đây:

DO [While (logical) Until (logical)]

•••••

Block of Code

LOOP [While (logical) Until (logical)]

Chỉ có một trong số bốn mệnh đề *While and Until* được sử dụng trong câu lệnh đúng. Sự thay thế của mệnh đề *While or Until* xác định khi giá trị lôgíc được kiểm tra. Việc đặt nó tại dỉnh của vòng lặp sẽ khiến cho nó được kiểm tra trước khi chạy vòng lặp cho lần đầu tiên. Việc đặt nó tại phần cuối sẽ khiến cho vòng lặp cần được tính toán ít nhất là một lần trước khi kiểm tra giá trị của câu lệnh lôgíc. Khi mệnh đề *While* được sử dụng, vòng lặp được thực hiện dẫu cho biểu thức lôgíc là TRUE, và dừng khi nó thay đổi thành FALSE. Mệnh đề *Until* chỉ là ngược, làm cho vòng lặp cần phải được thực hiện cho tới tận khi giá trị lôgíc trở thành TRUE.

Chẳng hạn, vòng lặp sau đây được tính toán cho tới khi giá trị của A tương đương với 10. Có lẽ có tính toán nào đó trong khối mã đang thay đổi giá trị của A và cuối cùng sẽ làm cho nó trở thành 10, mặt khác vòng lặp này sẽ chạy mãi.

```
DO Until (A = 10)
.....
Block of Code
.....
LOOP
```

Các câu lệnh For Each/Next là vòng lặp đặc biệt tạo ra cho các tập hợp của các đối tượng và các biến mảng. Tất cả các tên đa đối tượng là các tập hợp, chẳng hạn như WorkSheets và TextBoxes. Tập hợp WorkSheets chứa tất cả các đối tượng WorkSheet trong trang tính. Thông thường, bạn sẽ rút ra một bảng tính đặc biệt bằng cách áp dụng tên của bảng vào tập hợp WorkSheets. Nếu bạn muốn thực hiện một cái gì đó cho mỗi bộ phận của một tập hợp, hoặc mỗi thành phần của một mảng, hãy sử dụng vòng lặp For Each/ Next. Cú pháp của vòng lặp là:

```
FOR EACH object IN collection
.....
Block of Code
.....
NEXT object
```

Ö đây *object* là một đối tượng của kiểu được chứa trong *collection* và *collection* là tập hợp các đối tượng. *Collection* cũng có thể là một biến mảng - khi đó đối tượng sẽ là một thành phần của mảng. Vòng lặp tính toán một lần cho mỗi đối tượng trong tập hợp, với biến đối tượng lấy giá trị của một đối tượng khác khi mỗi đối tượng qua vòng lặp.

Ngoài ra, các vòng lặp là các câu lệnh *if.* Câu lệnh *if* kiểm tra giá trị lôgíc và thực hiện hoặc không thực hiện khối mã sau đây, dựa vào giá trị của giá trị lôgíc. Cú pháp của câu lệnh *if* khối là:

```
IF (logical1) THEN

....

Block of Code 1

....

ELSELF (logical2)

....

Block of Code 2

....

ELSE

....

Block of Code 3

....

END IF
```

Khi câu lệnh *if* khối được thực hiện đầu tiên, giá trị của *logical1* sẽ được kiểm tra. Nếu nó là TRUE, khi đó *Block of Code 1* được thực hiện. Nếu không, giá trị của *logical2* sẽ

được kiểm tra và nếu nó là TRUE, *Block of Code 2* được thực hiện. Có thể có nhiều mệnh đề *Elself* trong câu lệnh *if* khối để kiểm tra đối với các trường hợp khác nhau, nhưng chỉ có khối mã đầu tiên với giá trị lôgíc được thực hiện. Tất cả các khối mã khác được nhảy qua, dẫu cho các giá trị lôgíc của họ cũng là TRUE. Khối cuối cùng có thể là mệnh đề *Else*. Nếu không có giá lôgíc khác nào là TRUE thì *Block of Code 3* được thực hiện. Mệnh đề *Elself* và *Else* là tuỳ chọn.

Câu lệnh Select Case cho phép bạn lựa chọn một khối mã đơn lẻ từ một danh sách theo giá trị. Cú pháp là:

SELECT CASE value CASE compare 1 Block of Code 1 CASE compare 2 Block of Code 2 CASE ELSE Block of Code 3 END SELET

Đối số value được so sánh với comparel và nếu chúng phù hợp, Block of Code 1 sẽ được thực hiện. Nếu thay vào đó, compare 2 phù hợp thì Block of Code 2 được thực hiện. Nếu không có gì phù hợp thì mệnh đề Case Else tuỳ chọn được lựa chọn và Block of Code 3 được thực hiện. Chỉ có khối mã đầu tiên có giá trị so sánh phù hợp với đối số giá trị mới thực sự được thực hiện. Tất cả các khối mã khác đều được nhảy qua.

4.9.6. Gọi các thủ tục khác

Các thủ tục có thể là các chương trình độc lập, hoặc chúng có thể là một phần của các chương trình phân đoạn lớn hơn. Nếu một nhiệm vụ đặc biệt được thực hiện tại nhiều nơi trong một chương trình thì bạn có thể sao chép khối mã mà thực hiện nhiệm vụ đó vào mỗi vị trí này, hoặc bạn có thể tạo một thủ tục mới chứa khối mã đó và gọi thủ tục đó bất cứ khi nào cần. Việc phân đoạn các nhiệm vụ lặp lại làm giảm lượng mã bạn phải ghi và đơn giản hoá việc gỡ rối của mã đó. Ngoài ra, việc phân một chương trình thành các phần mà thực hiện các nhiệm vụ thẳng xuôi, đơn giản sẽ làm cho nó trở lên dễ hiểu hơn.

Để gọi một thủ tục Sub từ trong một thủ tục khác, bạn chỉ cần chèn tên của thủ tục ở nơi bạn muốn nhiệm vụ của nó được thực hiện. Nếu thủ tục có các đối số, hãy chèn chúng như một danh sách được tách biệt bằng dấu phảy sau tên thủ tục. Không nên bao quanh các đối số bằng các dấu ngoặc khi bạn gọi một thủ tục, tuy nhiên chúng được bao quanh bằng các dấu ngoặc trong tiêu đề mở đầu của thủ tục.

Chú ý: Bạn cũng có thể gọi một thủ tục bằng cách sử dụng câu lệnh CALL. Từ khoá CALL đứng trước tên thủ tục và các đối số của nó. Trong trường hợp này, các đối số phải được các dấu ngoặc đơn bao quanh.

Để gọi một thủ tục hàm hoàn toàn theo cách tương tự như khi gọi một trong số các hàm có sẵn. Ở vế phải của công thức, gõ tên thủ tục đứng trước dấu ngoặc đơn trái, đứng trước các đối số thủ tục, đứng trước dấu ngoặc đơn phải. Kết quả của thủ tục hàm được trở lại theo tên thủ tục và cũng có thể theo các đối số của thủ tục.

4.9.7. Tạo các hộp thoại

Các hộp thoại chứ không phải những hộp có sẵn được tạo trên bảng thoại. Khi bạn chèn một bảng thoại vào trong một trang tính bằng cách sử dụng lệnh *Insert > Macro > Dialog*, bạn sẽ thấy một hộp thoại mới và thanh công cụ Forms. Mỗi một nút trên thanh công cụ *Forms* tạo một đối tượng khác trên hộp thoại mới. Hộp thoại mới (*come with*) các nút OK và *Cancel* đã được chèn.

a) Kéo một hộp thoại:

Bạn tạo các đối tượng thêm vào trên hộp thoại bằng cách nhấn vào một trong các nút trên thanh công cụ Forms và sau đó kéo đối tượng trên hộp thoại. Một khi đối tượng được kéo trên hộp thoại, thì bạn có thể soạn thảo các đặc tính của đối tượng để thay đổi hình dáng của nó và việc nó thao tác ra sao. Các đặc tính được thay đổi nhờ lựa chọn hoặc soạn thảo trực tiếp đặc tính trên đối tượng, như với văn bản tựa đề, bằng cách sử dụng lệnh *Format > Object* hoặc nút *Control Properties* trên thanh công cụ *Forms*; hoặc bằng cách sử dụng hộp *Name* trên thanh công thức. Ngoài ra, nhiều đặc tính có thể được thay đổi theo chương trình điều hành, bằng cách thiết đặt giá trị của đặc tính có câu lệnh gán.

Chú ý: Việc gọi tên các đối tượng trên một hộp thoại phải được thực hiện bằng hộp Name trên thanh công thức. Điều này làm thay đổi tên thực của đối tượng. Nếu bạn sử dụng lệnh Insert > Name > Define, bạn sẽ tạo một tên mà có liên quan tới tên của đối tượng thay vì thay đổi chính tên của đối tượng.

Mỗi nút trên thanh công cụ *Forms* tạo ra một đối tượng mới hoặc cho phép bạn soạn thảo hoặc kiểm tra hộp thoại.

| Label | Nhãn là văn bản không thể được soạn thảo. Nó thường được sử dụng để chú thích một hộp thoại, hoặc để gửi một thông báo tới người dùng |
|-----------|---|
| Edit Box | Một hộp ở nơi mà người dùng có thể nhập thông tin. Kiểu thông in mà hộp chấp nhận được thiết đặt bằng lệnh Format > Object, tab Control |
| Group Box | Hộp nhìn bề ngoài dùng để nhóm các điều khiển. Nó cũng xác định nhóm điều khiển cho các nút lựa chọn |

| Create Button | Nút lệnh để khởi động một thủ tục. Một nút lệnh là nút ngầm định, nó được nhấn nếu người dùng nhấn Enter. Còn nút kia là nút Cancel, nó được nhấn nếu người dùng nhấn Esc. Có thể chỉ có một ngầm định và một nút Cancel trên hộp thoại | | | | |
|-------------------------------|--|--|--|--|--|
| Check Box | Một điều khiển để thiết đặt các lựa chọn bằng cách nhấn vào hộp | | | | |
| Option putton | Một điều khiển để lựa chọn một trong số tập hợp các lựa chọn loại trừ nhau. Việc nhấn vào một nút lựa chọn trong nhóm sẽ bác bỏ tất cả các nút khác | | | | |
| | Một nhóm gồm tất cả các nút lựa chọn được trực tiếp gắn vào hộp thoại, hay tất cả các nút lựa chọn được gắn vào Group Box đơn lẻ | | | | |
| List Box | Một hộp chứa danh sách các giá trị mà từ đó người dùng có thể lựa chọn một giá trị | | | | |
| Drop-Down | Hộp danh sách với một danh sách không được hiển thị cho tới khi người dùng nhấn vào nó. Chỉ có lựa chọn hiện thời mới được hiển thị | | | | |
| Combination List - Edit | Tổ hợp của List Box và Edit Box. Người dùng có thể kéo xuống một danh sách và lựa chọn một giá trị, hoặc gõ giá trị trong Edit Box | | | | |
| Combination Drop-Down Edit | Tổ hợp Drop-Down và Edit Box. Người dùng có thể kéo xuống một danh sách và lựa chọn một giá trị, hoặc gõ giá trị trong Edit Box | | | | |
| Scroll Bar | Thanh cuộn chuẩn. Nó có một giá trị, phụ thuộc vào vị trí của ngón tay cái (con trượt) | | | | |
| Spinner | Điều khiển mà giá trị của nó thay đổi theo mỗi lần bạn nhấn vào nó | | | | |
| Control Propertries | Hiển thị hộp thoại Format Object cho điều khiển lựa chọn | | | | |
| Edit Code | Hiển thị thủ tục đã gán cho đặc tính OnAction của điều khiển hoặc tạo một thủ tục mới nếu thủ tục không được gán | | | | |
| Toggle Grid | Bật và tắt lưới nền trên bảng Dialog. Nếu một bảng tính là có thể nhìn thấy được, nó cũng sẽ bật và tắt lưới bảng tính | | | | |
| Run Dialog | Khởi động cho Visual Basic thực hiện, có hộp thoại hoạt động. Việc nhấn một nút chạy thủ tục được gán thay vì lựa chọn nút | | | | |
| | Sử dụng để kiểm tra hộp thoại | | | | |

Chú ý: Đặc tính OnAction chứa tên của thủ tục sẽ thực hiện khi điều khiển được sử dụng.

b) Hiển thị hộp thoại:

Khi bạn kéo hộp thoại, cần một cách hiển thị nó bằng một chương trình chạy. Hộp thoại được hiển thị bằng cách sử dụng phương pháp Show của đối tượng hộp thoại. Cú pháp như sau:

SHEETS ("SheetName"). SHOW

Ở đây Sheet Name là tên của bảng thoại.

c) Làm ẩn một hộp thoại:

Khi đã thực hiện nhờ hộp thoại, bạn cần thoát ra khỏi nó để có thể tiếp tục với bất cứ cái gì chương trình của bạn còn đang thực hiện. Để ẩn hộp thoại, hãy sử dụng phương pháp *Hide*, hoặc thiết đặt các đặc tính *Dismiss* của nút trên hộp thoại. Phương pháp *Hide* thường được chèn trong thủ tục đã gán vào nút OK của hộp thoại. Cú pháp của phương pháp *Hide* như sau:

SHEETS ("SheetName"). HIDE

Đặc tính Dismiss của nút lệnh sẽ đóng hộp thoại nếu nút được nhấn.

d) Gán các thủ tục cho các điều khiển của hộp thoại:

Để giúp cho hộp thoại làm một cái gì đó, bạn cần gán các thủ tục cho các điều khiển trên hộp thoại. Bạn thực hiện điều này bằng đặc tính *OnAction*. Đặc tính *OnAction* của điều khiển chứa tên của thủ tục để thực hiện khi điều khiển được truy nhập. Tác dụng mà khởi động đặc tính phụ thuộc vào điều khiển. Một nút lệnh sẽ khởi động đặc tính khi nó được nhấn. Hộp soạn thảo khởi động đặc tính khi các nội dung của nó thay đổi.

Để thiết đặt thủ tục được gán cho đặc tính OnAction, hãy lựa chọn điều khiển, chọn lệnh Tools > Assign Macro, và lựa chọn thủ tục để chạy từ danh sách.

Sử dụng thủ tục tương tự để thay đổi các nội dung của đặc tính OnAction. Đặc tính OnAction cũng có thể được thay đổi nhờ chương trình đang chạy bằng cách sử dụng một câu lệnh giống như sau:

SHEETS ("SheetName"). ONACTION = ProcedureName

Ô đây SheetName là tên của bảng thoại và ProcedureName là tên của thủ tục sẽ chạy.

Chú ý: Một thủ tục đã gán cho một điều khiển thường được đặt tên bằng tên của điều khiển và tác dụng. Chẳng hạn, Button I_Click là tên của thủ tục được thực hiện bằng cách nhấn Button I.

4.10. TẠO CÁC THỦ TỤC HÀM

Khi bạn xác định một thủ tục, tiêu đề mở đầu của thủ tục sẽ xác định thủ tục Sub hoặc thủ tục Function. Ở nơi các thủ tục Sub thường thực hiện các tác dụng, các hàm Function trở về các giá trị. Điều đó không có nghĩa là thủ tục Function không thể thực hiện một tác dụng, duy chỉ có vậy nó cũng chỉ có thể trở về một giá trị và được sử dụng trong công thức. Kiểu thủ tục hàm đơn giản nhất và phổ biến nhất trở về một giá trị đơn; Tuy nhiên, các thủ tục hàm có thể trở về một mảng các giá trị, và chúng được xem xét theo cách tương tự như bất cứ hàm Excel nào khác mà trở về mảng.

4.11. GÕ RỐI CÁC CHƯƠNG TRÌNH MẠCRO

Trong nhiều trường hợp, các chương trình Visual Basic không làm việc một cách suôn sẻ vào lần đầu tiên. Để tìm được lỗi sai của mình, bạn cần xác định cái mà thủ tục đang thực hiện tại mỗi bước. Nếu bạn mắc lỗi thời gian chạy theo kiểu đó, thì hộp thoại lỗi sẽ xuất hiện cùng với lựa chọn để kết thúc, tiếp tục, bật lên đối với cửa sổ gỡ rối, hoặc nhảy đối với câu lệnh đã gây ra lỗi.

Trong hầu hết các trường hợp, bạn sẽ bật của số Debug lên.

Cửa số *Debug* hiển thị mã trong cửa sổ mã tại đáy. Câu lệnh đã gây ra lỗi được tạo viền. Trong trường hợp này, bạn đã được nói qua về câu lệnh *On Error* mà có khả năng bẩy lỗi. Nếu không có bẩy lỗi, bạn sẽ mắc lỗi thời gian chạy. Tại điểm này, có thể lựa chọn một biến và nhấn vào nút *Instant Watch* (trên thanh công cụ Visual Basic) để biết giá trị của biến. Chẳng hạn, nếu bạn lựa chọn biến Err, và nhấn vào *Instant Watch*, sẽ thấy rằng nó có giá trị 1006. Bạn có sự lựa chọn trong việc làm cho lựa chọn hiện thời trở thành biến đang theo dõi bình thường bằng cách nhấn vào nút Add trên hộp thoại *Instant Watch*.

Biến đang theo dõi bình thường xuất hiện trong ô đồng hồ tại đỉnh của của số *Debug*. Nhấn vào tab *Watch* trong ô *Immediate* là có thể nhìn thấy được. Biến theo dõi sẽ luôn luôn hiển thị giá trị hiện thời của biến đang theo dõi trong ngữ cảnh đã chỉ ra.

Ô *immediate* của cửa sổ *Debug* được dùng để thực hiện ngay lập tức bất cứ lệnh Visual Basic nào hoặc câu lệnh mà có thể được đặt trên một dòng dơn. Nhiều câu lệnh có thể được xếp vào ngăn bằng cách tách chúng nhờ dấu hai chấm.

Chú ý: Ô Immediate là một chỗ tốt để thứ nghiệm bằng mã mới và các thủ tục. Chạy chương trình ngắn để thiết lập môi trường của bạn, biến kích thước, v.y...

Đặt một lệnh dừng hoặc điểm ngắt để làm cho mã dừng trước khi kết thúc. Khi đó, bạn có thể kiểm tra các thủ tục và câu lệnh mới mà không phải tạo ra một chương trình riêng cho chúng, và chỉ kiểm tra chúng trong mã đó.

Cách khác để phát hiện lỗi là chạy mã trong chế độ *Step*. Trong chế độ *Step*, chương trình thực hiện một dòng tại một thời điểm, do vậy bạn có thời gian để kiểm tra các biến để biết chúng đang thay đổi ra sao, hãy chính lên điểm mà chương trình hỏng. Để nhanh chóng nhảy tới vùng xảy ra vấn để, hãy lựa chọn câu lệnh trong vùng đó và nhấn nút *Toggle Breakpoint*. Khi bạn chạy mã của mình, nó chạy ở tốc độ bình thường cho tới khi nó đạt tới điểm dừng, dừng lại và hiển thị cửa số Debug. Khi đó, bạn có thể tiếp tục chạy chương trình, mỗi lần một bước.

Chương 5

PHÂN TÍCH CÁC SỐ LIỆU THÍ NGHIỆM

Các số liệt thực nghiệm là một trong những nguồn tài liệu gốc quan trọng nhất của các kỹ sư và nhà khoa học. Tuy nhiên chúng ta thường gặp nhiều khó khăn trong việc quản lý và phản tích khối lượng khổng lồ và phức tạp của các số liệu thực nghiệm này. Excel cung cáp nhiều phương pháp để giúp chúng ta nhập và xử lý các số liệu đó. Trong chương này, bạn đọc sẽ làm quen với cách nhập các số liệu vào Excel, cách xử lý chúng bằng các công cụ của Excel một cách thuận lợi và nhanh chóng, chính xác.

5.1. NHẬP SỐ LIỆU THÍ NGHIỆM LÊN BẢNG TÍNH

Công việc phân tích các số liệu thực nghiệm có thể đơn giản chỉ là sắp xếp các số liệu theo một trật tự nào đó hoặc có thể phức tạp hơn là tìm kiếm từ một lượng dữ liệu rất nhiều để rồi lọc ra vài dữ liệu cần thiết theo các tiêu chuẩn cụ thể nào đó. Để Excel thực hiện giúp công việc này một cách nhanh chóng thì trước hết chúng ta phải nhập được dữ liệu vào trong bảng tính của Excel. Sau đó chúng ta có thể dùng các lệnh và các hàm sẵn có của Excel để tính trị số trung bình, tính các đặc trưng thống kê khác, lập hàm xấp xỉ hồi quy, về các đó thị biểu diễn quan hệ.

Có vài cách thức để nhập số liệu vào trong một bảng tính Excel bạn sẽ tuỳ chọn chúng theo dạng số liệu thực nghiệm cụ thể mà bạn đang có. Nếu đó là bản chép tay thì chỉ có cách gõ dắn các số liệu vào từ bàn phím. Nếu các số liệu đã có sẩn trong tệp trên đĩa mềm chẳng hạn dưới một dạng thức nào đó thì bạn có thể chuyển đổi dạng thức ghi đó để nhập vào bảng tính. Khi đó cứ gọi lệnh *Open* để mở tệp rồi căn chỉnh lại các cột chứ không cán gõ lại từ bàn phím nữa.

5.1.1. Nháp số liệu từ bàn phím

Thông thường các số liệu được góp nhặt từ các bản ghi chép tay. Muốn nhập vào bảng tính chỉ có cách gõ từ bàn phím. Nếu bạn có máy quét (scaner) và một phần mềm xử lý bản ghỉ ký tự quang học thì có thể nhập nhanh được các bản in số liệu (chứ không phải bản chép tay) rồi biến chúng thành một tệp trên đĩa đạng *Text* có thể đọc được. Nhưng sau đó bạn phải kiểm tra lại cẩn thận bằng cách so sánh bản in với bản kết quả của máy quét để tin chắc là chúng hoàn toàn giống nhau về nội dung. Phần mềm quét thông dụng ở Việt Nam hiện nay là TextBriđge (cho văn bản tiếng Anh) và VNDOC (cho văn bản tiếng Việt).

Nếu bạn phải tự gõ bàn phím thì sau đây là vài thao tác để gõ nhanh số liệu - nếu số liệu vào từng cột theo trật tự dòng 1 rồi dòng 2, v.v... Bạn hãy chọn cột bằng nhấn chuột sau đó gõ số liệu vào dòng thứ 1, bấm phím *Enter* để con trỏ tự xuống dòng thứ 2, gõ số liệu tiếp, lại nhấn *Enter*. Cứ làm mãi đến khi nhập xong hết số liệu của cột đã chọn. Nếu bạn nhập số liệu theo từng dòng thì nhập số liệu vào cột thứ 1, nhấn Tab thì con trỏ sẽ nhảy sang cột bên phải để bạn gõ tiếp số liệu. Cứ như thế mãi cho đến hết.

5.1.2. Nhập số liệu từ đĩa

Thông tin đã lưu trư trên đĩa dưới một số dạng như dã liệt kê trong bảng 5.1 sẽ dễ dàng được Excel chấp nhận. Ngoài ra Excel cũng có thể dọc một số dạng dữ liệu khác nữa khi dùng công cụ *Query add-in*. Dữ liệu sau khi đã được *Query add-in* xử lý sẽ dễ dàng đọc được trong Excel.

Các mô hình lập trên máy tính của các hiện tượng vật lý thường cho ra rất nhiều dữ liệu. Chúng cần được phân tích rồi vẽ đồ thị quan hệ. Nếu dữ liệu đã được ghi dưới dạng Text thì Excel sẽ đọc rất dễ dàng.

Nếu số liệu trên đĩa được ghi theo mã nhị phân thì bạn cần phải viết một chương trình nhỏ bổ trợ để đọc tệp nhị phân đó rồi ghi lại lên đĩa dưới dạng Text. Để chuyển đổi dễ dàng thì số liệu trong tệp dạng Text cần ghi thành bảng với các cột dữ liệu cách nhau bởi dấu Tab.

Excel cũng có thể đọc các tệp dạng Text mà các dữ liệu cách nhau bởi dấu phẩy, hoặc dấu cách (khoảng trống), hoặc bất cứ ký tự đơn lẻ nào, nhưng bạn phải khai báo trước đó trong hộp thoại *Import Wizard*.

| Dạng của số liệu | Tên đuôi của tệp trên đĩa |
|--------------------------------|---|
| EXCEL 5.0 hoặc 7.0 | .XLS,.XLT |
| EXCEL 4.0 | .XLS,.XLC,.XLM,.XLW,.XLA,.XLT,.XLB,.XLL |
| Văn bản (Text) | .T XT |
| Số liệu cách nhau bởi dấu phẩy | .CSV |
| Lotus 123 đời 3.X | .WK3,.FM3,.PRN |
| Microsoft Work | .WKS |
| Allways | .ALL |
| Data interchange | .DIF |
| dBASE II | .DBF 2 |
| dBASE III | .DBF 3 |
| dBASE IV | .DBF 4 |
| Quatro Pro for MS-DOS | WQ1,.WQS |

Bảng 5.1. Các dạng tệp dữ liệu mà Excel đọc được

5.1.2.1. Nhập các tệp dạng không hạn chế

Nếu dữ liệu đang ở dạng tệp Tab-delimited Text bạn chỉ cần mở tệp bằng lệnh *File | Open*. Các số liệu được phân cách bởi ký tự Tab và dấu cuối dòng chính là dạng tệp văn bản mặc định của Excel. Khi bạn mở tệp này thì Excel sẽ mở một sổ tính (Workbook) mới và đặt các ký tự từ tệp vào ô A1 cho đến khi nó gặp dấu Tab đầu tiên. Sau đó Excel sẽ di chuyển ô hoạt động sang bên phải một ô rồi đặt các ký tự tiếp theo dấu Tab vào ô B1 cho đến khi nó gặp dấu Tab tiếp theo. Excel lại di chuyển ô hoạt động sang bên phải một ô cho đến khi gặp dấu xuống dòng thì chuyển đến dòng dưới của cột A.

Nếu số liệu được phân cách bởi các dấu hiệu khác, ví dụ như dấu phẩy, dấu cách, bạn hãy chọn delimited từ danh sách liệt kê bởi Text Import Wizard sẽ xuất hiện khi mở tệp.

5.1.2.2. Nhập các tệp dạng Spark Gaps

Xét một ví dụ tệp văn bản như hình 5.1 có chứa các số liệu được tạo ra bởi phần mềm Windows Notepad. Bạn hãy làm theo các thao tác sau:

1. Dùng Notepad tạo ra tệp dạng Text như hình 5.1. Dùng phím Space để dóng thẳng các cột. Ghi lại trên đĩa với tên HINH5-0.TXT

2. Khởi động Excel và mở tệp trên bằng lệnh File/Open.

| 🕻 HINH5-0 - Notepad | | | | | | | | |
|--|---|--------|------|-------|-----------|------------|------|---|
| File Edi | t | Format | View | Help | | | | |
| Dữ liệu dạng cột Tệp tin mẫu để đọc vào excel | | | | | | | | |
| Điện áp | , | | 1 | Đường | kính điệi | n cực (arr | 1) | 1 |
| Điện áp | , | | 1 | | | | - | 1 |
| Điện áp |) | | 1 | 2,5 | - 5 | 10 | 20 | 1 |
| 5 | | | | 0.13 | 0,15 | 0.15 | 0.16 | 1 |
| 10 | | | | 0.27 | 0.29 | 0.30 | 0.32 | 1 |
| 15 | | | | 0.42 | 0.44 | 0.46 | 0.48 | 1 |
| 20 | | | | 0.58 | 0.60 | 0.62 | 0.64 | 1 |
| 25 | | | | 0.76 | 0.77 | 0.78 | 0.81 | 1 |
| 30 | | | | 0.95 | 0.94 | 0.95 | 0.98 | 1 |
| 35 | | | | 1.17 | 1.12 | 1.12 | 1.15 | 1 |
| 40 | | | | 1.41 | 1.30 | 1.29 | 1.32 | 1 |
| 45 | | | | 1.68 | 1.50 | 1.47 | 1.49 | 1 |

Hình 5.1: Nội dung tệp mẫu ví dụ dạng Text được tạo ra bởi Notepad.

3. Trong hộp thoại Open, hãy thay đổi nội dung hộp Files of Type thành All Files

4. Chon HINH5-0.TXT rồi nhấp chuột lên Open

Khi đó trên màn hình hiện ra Text Import Wizard để giúp bạn nhập tệp vào Excel.

5. Nhấp nút Delimited rồi Next.

6. Trong bước thứ hai của Wizard, chọn hộp kiểm tra Space và hộp kiểm tra Treat Consecutive Delimiters As One. Nhấp vào Next để tiếp tục.

D÷ liÖu d'ng cét TÖp tin mấu ©Ó ©ặc vụo excel 1 §-êng kÝnh ©iÖn cùc (cm) §iÖn "p 1 ----§iÖn 'p _____ _ _ • Þ ◀ Next > < <u>B</u>ack Einish Cancel

98

7. Trong bước thứ ba của *Wizard*, bạn sẽ thiết lập định dạng của dữ liệu trong mỗi cột. Chọn *General* để Excel định dạng các cột theo dạng văn bản và con số. Nhấp vào *Finish* để tiếp tục.

| Text Import Wizard - Step 3 of 3 | ?× | | | | | | |
|---|----------------------|--|--|--|--|--|--|
| This screen lets you select each column and set the Data Format. 'General' converts numeric values to numbers, date values to dates, and all remaining values to text. <u>Advanced</u> | | | | | | | |
| Data preview | | | | | | | |
| GenerGenerGenerGeneral | GenerGenerGenera | | | | | | |
| D÷ liÖu d¹ng cét TÖp tin nBu ⊅Ó | שמכ אוס excel | | | | | | |
| §iÖn .p 1 §-êng §iö .p 1 | kÝnh SiÖn cùc 1 | | | | | | |
| 4 | > | | | | | | |
| Cancel | < Back Next > Finish | | | | | | |

Bây giờ trang tính sẽ như hình 5.2. Excel đã có thể phân cách các dữ liệu số, nhưng các chữ tiêu đề thì còn chưa gọn. Bạn phải tự mình sửa lại cho đẹp mắt.

| | Arial • 10 • B I U ≣ ≣ ≣ 혐 \$ % , 100 ₽ ₽ ₽ | | | | | | | | |
|----------|---|------|-------|-------|------|------|-------|------|---|
| | G14 | • | f* | | | | | | |
| | A | 8 | C | D | E | F | G | Н | |
| 1 | Dữ | liệu | dang | cột | | | | | |
| 2 3 | Tệp | tin | กออับ | để | ರ್ಧೆ | vao | excel | | |
| 4 | Điện | áp | 1 | Đường | kính | điện | cực | (cm) | 1 |
| 5 | Điện | áp | 1 | | 1 | | | | |
| 6 | Điện | áp | 1 | 2,5 | 5 | 10 | 20 | 1 | 1 |
| 7 | 5 | 0.13 | 0,15 | 0.15 | 0.15 | 1 | | | |
| 8 | 10 | 0.27 | 0.29 | 0.3 | 0.32 | 1 | | | |
| 9 | 15 | 0.42 | 0.44 | 0.46 | 0.48 | 1 | | 4 | |
| 10 | 20 | 0.58 | 0.6 | 0.62 | 0.64 | 1 | | | |
| 11 | 25 | 0.76 | 0.77 | 0.78 | 0.81 | 1 | | | |
| 12 | 30 | 0.95 | 0.94 | 0.95 | 0.98 | 1 | | | |
| 13 | 35 | 1.17 | 1.12 | 1.12 | 1.15 | 1 | | | |
| 14 | 40 | 1.41 | 1.3 | 1.29 | 1.32 | 1 | | | |
| 15 16 | 45 | 1.68 | 15 | 1.47 | 1.49 | 1 | | | |

.

99

Sau khi sửa chữa, chỉnh sữa bảng nhập từ Notepad vào, ta có được bảng trong excel như trong hình 5.2:

| | · Arial | - | • 12 • | <u>в / U</u> | 돌 물 콜 |
|----|---------|-----------|---------------|--------------|-------|
| | E9 | • | ∲ 0.32 | | |
| | A | В | C | D | 國際国家商 |
| 1 | | DÙLIÈUI | DANG CÓT | | |
| 2 | | TEP TIN N | HÁU ĐỂ ĐO | C VÀO EX | CEL |
| Э | | | | | |
| 4 | Điện áp | Đu | rờng kính d | fiên cực (d | cm) |
| 5 | dinh | | | | |
| 6 | KV | 2,5 | 5 | 10 | 20 |
| 7 | | | | | |
| 8 | 5 | 0.13 | 0,15 | 0.15 | 0.16 |
| 9 | 10 | 0.27 | 0.29 | 0.3 | 0.32 |
| 10 | 15 | 0.42 | 0.44 | 0.46 | 0.48 |
| 11 | 20 | 0.58 | 0.6 | 0.62 | 0.64 |
| 12 | 25 | 0.76 | 0.77 | 0.78 | 0.81 |
| 13 | 30 | 0.95 | 0.94 | 0.95 | 0.98 |
| 14 | 35 | 1.17 | 1.12 | 1.12 | 1.15 |
| 15 | 40 | 1.41 | 1.3 | 1.29 | 1.32 |
| 16 | 45 | 1.68 | 1.5 | 1.47 | 1.49 |

Hình 5.2: Nội dung trang tính EXCEL sau khi đã sửa chữa cho đẹp.

5.1.2.3. Nhập các tệp có chứa dữ liệu dạng bảng

Khi bạn có các dữ liệu đã ghi sẵn trên đĩa theo dạng các cột số liệu thì Excel có thể tải chúng vào và sắp xếp các dữ liệu vào các cột riêng biệt mà không cần phải thay đổi khoảng cách giữa các dấu Tab. Rất nhiều ứng dụng của các chương trình chạy trong DOS thường cho ra kết quả dưới dạng các bảng trong tệp văn bản (Text), ví dụ như chương trình SAP-90 để tính toán kết cấu mà các kỹ sư xây đựng ở nước ta thường quen dùng. Khi đó bạn hãy dùng kỹ thuật phân cách các cột với chiều rộng cố định của dữ liệu vào các cột của trang tính Excel.

Hãy dùng công cụ Text Import Wizard khi bạn mở tệp dữ liệu hoặc dùng công cụ Convert Text To Columns Wizard bằng cách gọi lệnh Data / Text To Columns. Tuy nhiên nếu tệp dữ liệu không được ghi với các khoảng cách đơn thì cách làm này không hiệu quả. Khái niệm "Ghi với khoảng cách đơn" có nghĩa là mỗi chữ có chiều rộng giống nhau, điều này đúng với hầu hết các chương trình chạy trong DOS. Còn trong Windows thì hầu hết các font chữ lại không như vậy chúng được ghi với khoảng cách tỷ lệ thuận với bề rộng văn bản, ví dụ chữ i sẽ hẹp hơn chữ w.

Nếu bạn tải một tệp văn bản vào Excel mà không thực hiện kỹ thuật nói trên thì các dòng văn bản từ tệp sẽ được xếp trong các ô lần lượt của cột A trong trang tính. Mỗi ô của cột sẽ chứa nguyên một dòng văn bản.

5.1.2.4. Nhập các tệp có chứa dữ liệu dạng bảng với con số và chữ lẫn lộn

Ví dụ bạn có tệp như hình 5.3 khá phức tạp bao gồm cả các văn bản, con số, v.v... Bạn hay mở tệp và dùng tuỳ chọn *Fixed Width* trong công cụ *Text Import Wizard* hay mở tệp như là tệp văn bản và dùng lệnh *Data/Text To Columns*. Nó sẽ mở ra một *Wizard* với các tuỳ chọn giống như của *Text Import Wizard*. Để hiểu rõ hơn, xin bạn hãy thử làm theo cách của ví dụ sau:

1. Tạo một tệp dạng *Text* như hình 5.3 bằng *Notepad* rồi ghi trên đĩa với tên tệp là HINH5-3.TXT chẳng hạn:

| 🛱 HINH5-3 - Notepad | | | | | | | | | |
|---------------------|---|--------|------|---------|----------|-----------|--|--|--|
| File Edite I | Format View | Help | | | | | | | |
| Các hạt cơ | Các hạt cơ bản: tệp văn bản mẫu để nhập vào EXCEL | | | | | | | | |
| Ho hat | Ten Hat | Mass | Spin | Strange | Charge | LifeTime(| | | |
| * | | | | | | | | | |
| Electron | Photon | 0 | 1 | 0 | 0 | infinite | | | |
| | Electron | 1 | 1/2 | - | - e | infinite | | | |
| | Neutrino | 0 | 0.5 | - | 0 | infinite | | | |
| | Muon | 206.77 | 0.5 | | -1.6e-19 | 2.212e-16 | | | |

Hình 5.3: Các hằng số vật lý của vài hạt cơ bản, ví dụ bảng do Notepad tạo ra.

2. Khởi động Excel, mở tệp HINH5-3.TXT. Trong bước thứ 1 của Text Import Wizard, hãy nhấp vào tuỳ chọn Fixed Width rồi nhấn vào Next để tiếp tục.

| Text Import Wizard - Step 1 of 3 | ?× |
|---|----------|
| The Text Wizard has determined that your data is Fixed Width. If this is correct, choose Next, or choose the data type that hest describes your data | |
| Original data type | |
| Choose the file type that best describes your data: | |
| Delimited - Characters such as commas or tabs separate each field. Fixed width - Fields are aligned in columns with spaces between each field. | |
| Start import at row: 1 🚖 File origin: 932 : Japanese (Shift-JIS) | <u>·</u> |
| Preview of file D:\Ho So Ca Nhan\ThuDinh\Nghiencuu\vietsach\excel\HINH5-3.txt. | |
| 1 COc hOt cO bOn: tOp vOn bOn mOu OO nhOp vOo EXCEL 2 do hat Ten Hat Mass Spin Strang | ge |
| 4 Blectron Photon 0 1 | |
| 5 Blectron 1 1/2 | |
| 4 | <u>}</u> |
| Cancel < Back Next > E | inish |

Hình 5.4: Bước thứ 1 của Text Import Wizard.

3. Trong bước thứ 2 của *Text Import Wizard*, hãy chèn các dòng giới hạn để đánh dấu nơi mà Excel sẽ bẻ gãy văn bản thành các cột. Nhấn vào *Next* để tiếp tục.

| This screen lets y | ou set field wit | iths (colu | ımn break | s). | | | |
|---|---|---------------------------------------|-----------------------------------|----------------------|----------------|---------------------|---|
| Lines with arrow | vs signify a colu | umn brea | k. | 10.40 | | | |
| To CREATE a To DELETE a To MOVE a b | i break line, clic break line, dou reak line, click a | k at the o ble click o and drag | desired pa on the line it . | sition. | | | |
| Data preview | 0 | 20 | | 80 | 40 | 50 | |
| Cac hat co Ho hat | ban : te Ten Hat | p van Mass | ban m Spin | au de nha Strange | p vao Charg | EXCEL e LifeTime | |
| Electron | Photon | 01 | 1 | 0 | 0 | Infinite | X |

Hình 5.5: Bước thứ 2 của Text Import Wizard.

4. Trong bước thứ 3 của Text Import Wizard, hãy lại định dạng lần nữa để chuyển đổi mỗi cột của dữ liệu, về định dạng General rồi nhấn Finish.

| 100 | A | B | C | - D | E S | F | G |
|-----|------------|----------|-------|-------|-----------|-------|---------------|
| 12 | Cac hat co | ban : te | p van | ban m | au de nha | p vao | EXCEL |
| 2 | Ho hat | Ten Hat | Mass | Spin | Strange | Charg | e LifeTime(s) |
| 3 | | | | | | | |
| 4 | Electron | Photon | 00 | 1 | 0 | 0 | Infinite |
| 3 | D Electro | n 1 | 2-Jan | - | #NAME? | Infi | nite |
| 6 | D Neutrin | o 0 | 0.5 | • | 0 | Infi | nite |
| 1 | Muon0 I | n 206 | .77 0 | 0.5 | 1.6e | -19 | 2.21E-16 |
| -8 | D Neutrin | 0 0 | 0.5 | - | 0 | Infi | nite |
| 9. | 0 | | | | | | |

Hình 5.6: Dạng ban đầu sau khi chuyển đổi bảng dữ liệu từ ngoài vào EXCEL.

5. Trong trang tính, hãy mở rộng cột A sao cho cho nhìn thấy toàn bộ văn bản. Bạn sẽ thấy các số liệu có thể nằm trong các cột nhưng các nhãn thì khá lộn xộn như hình 5.4. Hãy sửa lại bằng tay trên trang tính sao cho đúng nội dung, bạn sẽ được kết quả như hình 5.5. Bây giờ dữ liệu đã sẵn sàng để xử lý.

| - | | Contraction of the second second | and the second se | | | | |
|------|------------------|----------------------------------|---|--------|----------|-----------|-------------|
| | 唐田市▲ 4日21 | 戦略国家で | Cas | DN | E | F | G |
| | Cac hat co l | Dan | Tep var | han ma | u de nha | p vao EX | CEL |
| 2 | Ho hat | Ten Hat | Mess | Spin | Strange | Charge | LifeTime(s) |
| a'3+ | | | | | | | |
| 4 | Electron | Photon | 0 | 1 | 0 | 0 | Infinite |
| 5 | | Electron | 1 | 1/2 | - | -e | Infinite |
| б | | Neutrino | 0 | 0.5 | - | 0 | Infinite |
| 7 | Muon | Muon | 206.77 | 0.5 | | -1.60E-19 | 2.21E-16 |
| 8 | | Neutrino | 0 | 0.5 | - | 0 | Infinite |

Hình 5.7: Trang tính với các dữ liệu đã được nhập và sửa gọn ghẽ.

5.13. Nhập dữ liệu bằng Visual Basic for Application

Ngòn ngữ lập trình Visual Basic for Application chứa sẵn trong Excel có khả năng đầy đi để nhập bất cứ dạng dữ liệu nào vào trong trang tính của Excel, dù đó là tệp dạng văn bản hay tệp nhị phân. Bạn có thể viết các chương trình đặc biệt để đọc các tệp dữ liệu và sắp xếp chúng thành các cột. Sau đây là một lời khuyên nhỏ: trong thủ tục V3A để đọc tệp dữ liệu, hãy dùng câu lệnh *Line Input* để đọc dòng số liệu từ tệp vào bến chuỗi là strALine. Chuỗi này rồi sẽ được bẻ gẫy thành các mẩu bằng cách dùng tàm Mid\$().

5.14. Nhập dữ liệu từ cơ sở dữ liệu ngoài

Nếi dữ liệu nằm trong cơ sơ dữ liệu ngoài thì Excel có thể dùng công cụ Query (truy vấn) cể tiếp cận dữ liệu đó. Cài đặt Query vào trong Excel bằng chọn lệnh Tool/Add-in rồi kim tra hộp Query. Khi đã cài đặt xong thì lệnh Get External Data sẽ xuất hiện trên bảng chọn Data. Lúc đó bạn có thể khởi động Query, mở cơ sở dữ liệu ngoài, lập mặt chuyểi tiếp đồ hoạ để truy vấn rồi thực hiện truy vấn. Bộ phát cơ sở dữ liệu hoặc cơ sở dữ liệu ngoài sẽ thực hiện truy vấn và trả lại bộ dữ liệu đã được chọn lựa xong. Khi bạn đã nhin được bảng mà mình đang mong đợi, hãy quay trở về Excel rồi chên cơ sở dữ liệu này vào bảng tính của mình.

5.2. LƯU TRỮ VÀ KHAI THÁC CÁC DỮ LIỆU

Trong Excel chúng ta có thể dùng bất cứ vùng nào của bảng tính để làm cơ sở dữ liệu. Khi độ mỗi cột trong vùng được chọn sẽ là một trường, còn mỗi trường có thể chứa đến 255 ký tự (hoặc là văn bản, hoặc là con số, hoặc là công thức). Mỗi dòng trong vùng đó sẽ là một bản ghi của cơ sở dữ liệu. Mỗi bản ghi chứa các dữ liệu có liên quan đến nhau. Ví dụ các tham số đo được trong cùng một thí nghiệm sẽ có thể được lưu giữ trong các bản ghi riêng rẽ. Mỗi trường sẽ chứa các dữ liệu của tham số khác nhau như điện áp, áp suất.

Địrh dạng của cơ sở dữ liệu là giống như của bảng ghi chép dữ liệu. Điều đáng chú ý là Excel có các công cụ lệnh và các phương pháp hiệu quả để tiếp cận và xử lý khai thác cơ sở dữ liệu.

5.2.1. Tạo lập vùng dữ liệu

Bản ghi đầu tiên trong vùng dữ liệu của Excel sẽ chứa các tên trường. Chúng giống như tên các cột trong bảng ghi chép dữ liệu trên giấy. Bạn nên đặt tên trường sao cho đơn giản và gợi nhớ bản chất của các con số trong trường đó, tên này chỉ được ghi trong một dòng. Không được bỏ cách dòng bên dưới nó. Bên dưới dòng tiêu đề tên trường sẽ là lần lượt các bản ghi của vùng dữ liệu

5.2.2. Sử dụng các lệnh cơ sở dữ liệu

Có 5 lệnh trong bảng chọn *Data* được dùng để tìm kiếm và xử lý các bản ghi trong cơ sở dữ liệu:

- Lệnh Filter/Autofilter chọn ra bản ghi nào đáp ứng giá trị trong cột của cơ sở dữ liệu.

- Lệnh Filter/Show All tái hiện lại toàn bộ cơ sở dữ liệu sau khi nó đã được lọc bởi lệnh Filter/Autofilter.

- Lệnh Filter/Advanced Filter cho phép chúng ta tạo ra tiêu chuẩn phức tạp trên sổ tính rồi dùng tiêu chuẩn này để lựa chọn các bản ghi từ cơ sở dữ liệu.

- Lệnh Form biểu thị dạng để tìm kiếm và quản lý dữ liệu trong cơ sở dữ liệu.

- Lệnh Sort sắp xếp các dòng trong cơ sở dữ liệu theo một khoá nào đó.

5.2.2.1. Sử dụng khuôn mẫu dữ liệu

Cách dơn giản nhất để tiếp cận đến cơ sở dữ liệu trong Excel là dùng lệnh Data / Form, Excel dùng các tên trường trong dòng đầu tiên của vùng dữ liệu để tạo ra một khuôn mẫu dữ liệu để chứa các trường đó. Khuôn mẫu dữ liệu bao gồm các nút lệnh để thêm vào các bản ghi, hoặc để xoá bản ghi, để tạo ra tiêu chuẩn, và để tìm kiếm các bản ghi nào thoả mãn tiêu chuẩn đó. Nếu bạn muốn dùng khuôn mẫu, hãy nhấp chuột vào bất cứ chỗ nào trong vùng dữ liệu và chọn lệnh Data / Form. Khi đó hộp thoại Data Form sẽ hiện ra như hình dưới đây. Các trường sẽ được có trong hộp thoại tuỳ theo nội dung cụ thể của cơ sở dữ liệu của chính bạn.

5.2.2.2. Sử dụng lệnh Data/ Filter/ AutoFilter

Lệnh Data / Filter / AutoFilter chèn một danh sách kéo xuống ở đầu của mỗi cột (trường), trong đó chứa mọi giá trị tìm được trong cột này. Khi chọn một trong các giá trị đó hạn chế các bản ghi nhìn thấy được trong cơ sở dữ liệu để chỉ các bản ghi nào với giá trị có trong cột đã chọn. Mũi tên chỉ xuống của danh sách kéo xuống sẽ thay đổi màu sắc cho mỗi cột đang lọc dữ liệu cũng như số dòng của các dòng đã được chọn. Khi chọn giá trị khác tiếp theo sẽ hạn chế phần nhìn thấy được của cơ sở dữ liệu. Khi bạn chon muc *Custom* thì sẽ cho phép



Hình 5.8: Hộp thoại

bạn lựa chọn vùng các giá trị, còn nếu bạn chọn mục All thì sẽ lấy đi bất cứ gì đã lọc được ở cột đó. Để tắt AutoFilter, bạn hãy lại chọn Data/Filter/ AutoFilter một lần nữa.

5.2.2.3. Sử dụng lệnh Data/ Filter/ Show All

Lệnh Datal Filter/ Show All thể hiện lại toàn bộ cơ sở dữ liệu sau khi nó đã được lọc bằng lệnh AutoFilter.

5.2.2.4. Dùng lệnh Data/ Filter/ Advanced Filter

Lệnh Data/ Filter/ Advanced Filter cho phép chúng ta tạo ra tiêu chuẩn lọc phức tạp hơn cái mà tạo ra được bởi lệnh Data/ Filter/ AutoFilter. Tiêu chuẩn được tạo ra trên sổ tính trong vùng tiêu chuẩn. Bạn cũng có thể dùng lệnh Copy To range để trích dẫn các bản sao của bản ghi từ cơ sở dữ liệu. Khi bạn dùng lệnh này thì một hộp hội thoại xuất hiện cho bạn chỉ định Tiêu chuẩn cơ sở dữ liệu và Copy To range.

| Advanced | Filter | 1 |
|--------------------------|----------------|----------|
| Action Eilter the lis | t, in-place | OK |
| Copy to and | other location | Cancel |
| List range: | \$A\$1:\$D\$4 | |
| <u>C</u> riteria range: | <u>.</u> | |
| Copy to: | | |
| Unique record | ds only | |

Hình 5.9: Hộp thoại Advanced Filter.

Các tuỳ chọn là để lọc danh sách tại chỗ, giống như lệnh Data/ Filter/ AutoFilter hoặc để sao chép các bản ghi đã được chọn lựa sang một khu vực khác Nếu bạn dùng một vùng trích dẫn (copy To range) thì bạn cũng có thể hạn chế các trường được sao chép vào vùng trích dẫn đó.

5.2.2.5. Định nghĩa vùng tiêu chuẩn

Khi muốn dùng lệnh Datal Filter/ Advanced Filter bạn cần phải tạo ra vùng tiêu chuẩn. Vùng này bao gồm bản sao của bản ghi đầu tiên trong cơ sở dữ liệu có chứa các tên trường, cộng thêm với một hay nhiều dòng để chèn tiêu chuẩn tìm kiếm. Tiêu chuẩn tìm kiếm là các nhãn hay các giá trị để tìm kiếm, hoặc là biểu thức lôgic để tìm kiếm trong vùng của các giá trị. Bạn có thể đặt tiêu chuẩn tìm kiếm dưới bất cứ tên trường nào để thu hẹp hơn nữa việc tìm kiếm. Tiêu chuẩn đặt trong cùng dòng được giả thiết là nối với ý nghĩa lôgic AND. Tiêu chuẩn đặt trong các dòng khác nhau được giả thiết là được nối với ý nghĩa lôgic OR. Có 2 ký tự dùng để tìm kiếm các trường nhãn là: ? và *. Dấu

hỏi có thể đứng ở trước bất cứ ký tự đơn lẻ nào. Ví dụ, C?T sẽ tương ứng với CAT, hay COT, hay CUT. Còn dấu sao (*) tương ứng với bất cứ con số hoặc ký tự nào. Ví dụ C* tương ứng với *Common*, hay *Creation* hay *Continue*.

5.2.2.6. Định nghĩa vùng xuất kết quả

Nếu bạn dự định trích dẫn các bản ghi từ cơ sở dữ liệu thay cho việc đơn giản chỉ hạn chế các dòng nhìn thấy được tại chỗ, thì bạn phải đăng ký một vùng xuất kết quả. Vùng này bao gồm bản sao của bản ghi đầu tiên có chứa tên các trưởng (không cần mọi tên trường mà chỉ cần những tên trường nào mà bạn định trích dẫn).

Vùng xuất kết quả có thể gồm một hay vài dòng. Nếu vùng này chỉ có một dòng chứa các tên trường thì Excel sẽ xoá sạch mọi ô bên dưới các tên trường trước khi các lệnh cơ sở dữ liệu bắt đầu viết các bản ghi vào đó. Nếu có bất kỳ thông tin nào trong các ô bên dưới tên trường thì chúng sẽ bị mất. Nếu bạn định nghĩa vùng xuất kết quả gồm nhiều dòng thì các lệnh cơ sở dữ liệu sẽ viết các bản ghi vào đó cho đến khi chúng được ghi hết và rồi tạo ra một lỗi. Sử dụng các vùng xuất kết quả nhiều dòng là cách để bảo vệ dữ liệu quý giá của bạn bên dưới vùng xuất kết quả. Như vậy tốt hơn là nếu bạn đặt vùng này trong khu vực không có gì bên dưới nó nữa

5.2.2.7. Sắp xếp lại một danh sách

Bạn có thể dùng lệnh *Datal Sort* để sắp xếp lại một danh sách theo các giá trị trong một hay nhiều cột của danh sách đó. Để sắp xếp một danh sách, hãy chọn toàn bộ vùng cần sắp xếp rồi chọn lệnh *Datal Sort*. Nếu bạn đang sắp xếp cơ sở dữ liệu, đừng chọn dòng đầu tiên chứa các tên trường. Hộp thoại *Sort* có chỗ trống cho 3 khoá sắp xếp. Hãy chọn các cột mà bạn muốn sắp xếp và chọn trật tự mà bạn muốn sắp xếp chúng. Nếu bạn đang sắp xếp cơ sở dữ liệu, các trường được liệt kê bằng tên trường, nếu ngược lại thì chúng được liệt kê bằng tham chiếu đến các ô trong cột.

| Sort | • 12 • 10 | ? × |
|-------------|-------------------------|---------|
| Sort by | | |
| Ap suat | Ascending | |
| Then by | © <u>D</u> escending | |
| Nhiet do | | |
| These has | C Descending | 1 miles |
| Inen by | | |
| | | |
| My list has | | - |
| Header tow | 🔿 No header ro <u>w</u> | |
| Options | OK Can | cel |

Hình 5.10: Hộp thoại để chọn cách sắp xếp dữ liệu.

| | Name B | DX B | C | D |
|--|--------|---------|----------|--------|
| 1 | So TT | Ap suat | Nhiet do | Cao do |
| 2 | 3 | 10 | 31 | 220 |
| 3 | 1 | 12 | 30 | 200 |
| 4 | 2 | 13 | 32 | 210 |
| Contraction of the local division of the loc | | | | |

Hình 5.11: Kết quả một ví dụ cơ sở dữ liệu sau khi được sắp xếp theo cột B tăng dần.

5.2.3. Dùng các hàm xử lý dữ liệu

Các hàm thống kê cơ sở dữ liệu đã được liệt kê trong chương 1, thực hiện các phép tính thống kê trên các bản ghi nào thoả mãn tiêu chuẩn. Ví dụ bạn có thể tính kỳ vọng hoặc phương sai của dữ liệu trong các bản ghi nào có một vài đặc trưng nào đó.

Mỗi hàm dữ liệu có 3 đối số là: cơ sở dữ liệu, tiêu chuẩn và trường. Đối số cơ sở dữ liệu là một tham chiếu đến vùng cơ sở dữ liệu. Đối số tiêu chuẩn là một tham chiếu đến vùng tiêu chuẩn, còn đối số trường là tên trường trong cơ sở dữ liệu mà sẽ sử dụng các hàm thống kê sẽ hoạt động. Đối số trường có thể là văn bản hoặc tên trường từ dòng đầu tiên của cơ sở dữ liệu, hoặc số hiệu cột trong cơ sở dữ liệu (như vậy, cột đầu tiên là 1, cột thứ hai là 2, v.v...).

Chương 6 LẬP CÁC ĐƯỜNG CONG BIỂU DIỄN ĐỔ THỊ

Trong chương này trình bày về cách vẽ các đường cong dựa theo các điểm dữ liệu. Các phương pháp đơn giản nhất là sử dụng các hàm hồi quy tuyến tính của Excel: các hàm LINEST và LOGEST, hai hàm này thực hiện phép hồi quy tuyến tính bội. Bằng cách chuyển đổi một cách thích hợp các phương trình, bạn có thể làm xấp xỉ nhiều phương trình phi tuyến bằng các hàm này. Ngoài ra, có thể sử dụng hàm LINEST để thực hiện phép hồi quy đa thức cho việc làm xấp xỉ một tập dữ liệu bằng một đa thức. Đối với các phương trình phức tạp hơn, bạn sẽ sử dụng thuật toán hạ bậc nhanh nhất để tìm các hệ số cho một phương trình phi tuyến. Sau đó bạn sẽ tự động hoá thuật toán đó với Solver và chương trình Macro.

Khi việc làm xấp xỉ các dữ liệu thực nghiệm bằng một hàm đã biết là khó khăn hoặc không hợp lý, bạn có thể sử dụng các hàm tìm kiếm trong bảng và hàm nội suy. Bạn sẽ khai triển một hàm nội suy đơn giản trong bảng tính, và một hàm phức tạp hơn dưới dạng một hàm Macro.

Một dạng bài toán thông thường mà các nhà khoa học và các kỹ sư phải giải quyết trong thực tiễn sản xuất là phải tìm ra một phương trình giải tích có các giá trị xấp xỉ gần đúng so với các giá trị của một tập hợp các điểm dữ liệu. Ví dụ các nhà khoa học thường phải làm xấp xỉ một phương trình lý thuyết với các dữ liệu thực nghiệm nào đó để chứng minh một lý thuyết nào đó. Còn các kỹ sư thường phải làm cho các dữ liệu đo lường được trên các máy đo xấp xỉ với các giá trị có thể tính ra được theo một phương trình giải tích nào đó để họ có thể chuyển đầu ra từ công cụ đo này sang tham số vật lý đang được đo.

Loại bài toán này thường được gọi là bài toán tìm hàm số thực nghiệm và có thể giải bằng nhiều phương pháp mà trong đó quen thuộc nhất là phương pháp bình phương tối thiểu.

Đã có nhiều sách mô tả các chương trình mẫu theo các ngôn ngữ C hoặc Pascal hoặc Basic để thực hiện công việc này.

Trong chương sách này chúng ta sẽ bàn về khả năng của Excel để giải quyết công việc nói trên. Nói chung, Excel cung cấp 3 cách thức để lập ra và vẽ đường cong biểu diễn hàm số xấp xỉ theo các điểm giá trị thực nghiệm cho trước. Bạn có thể vẽ đường cong biểu diễn của hầu hết các phương trình với dữ liệu bằng các lệnh hồi quy tuyến tính

cài sẵn. Hoặc cũng có thể sử dụng phép hồi quy tuyến tính để vẽ đường cong dữ liệu phi tuyến bằng cách chuyển đổi dữ liệu một cách thích hợp trước khi vẽ. Ngoài ra, bạn còn có thể vẽ đường cong biểu diễn các phương trình phức tạp hơn bằng cách hiệu chỉnh bằng tay các hệ số của phương trình cho đến khi sai số dư (tổng bình phương của các sai số giữa dữ liệu thực và giá trị tính theo đường cong hàm số xấp xỉ) giảm đến mức tối thiểu, hoặc hệ sô tương quan tăng tới cực đại. Cuối cùng, trong trường hợp dữ liệu không thể lầm xấp xỉ được với bất cứ đường cong thích hợp nào thì bạn có thể sử dụng các hàm tìm kiếm trong bảng và phép nội suy để cung cấp các giá trị xấp xỉ.

6.1. SỬ DỰNG CÁC HÀM CÀI SẵN

Excel có khả năng dùng các đường cong cài sẵn (hàm số cài sẵn) để tạo ra các hàm xấp xỉ giống như phép hồi quy tuyến tính bội. Khả năng này cho phép tìm đường cong hàm xấp xỉ dữ liệu với một đường đơn hoặc với một đa thức phức. Bạn có thể hoàn thành phần lớn các công việc vẽ đường cong bằng khả năng hồi quy tuyến tính của Excel.

6.1.1. Các phép tính hồi quy

Khi làm xấp xỉ một đường cong với một số điểm dữ liệu nhờ sử dụng phép hồi quy, bạn sẽ làm giảm đến mức tối thiểu sai số bình phương số dư giữa các điểm dữ liệu và đường cong (giải tích bình phương nhỏ nhất). Sai số bình phương số dư (E) này được tính bằng phương trình sau:

$$E = \sum_{i=1}^{n} y(x_i) - (y_i)^2$$

 \dot{O} đây: $y(x_i)$ - phương trình đường cong đang được làm xấp xỉ;

n - số lượng các điểm dữ liệu;

x, và y, - các toạ độ của các điểm dữ liệu.

Excel sử dụng phép hồi quy tuyến tính bội, vì vậy nó giả thiết đường cong $y(x_i)$ có dạng:

$$y(x_{1,i}, x_{2,i},...) = A + Bx_{1,i} + Cx_{2,i} + ...$$

Trong đó: A, B và C - các hệ số của phương trình cần được hiệu chỉnh để làm cho đường cong xấp xỉ với dữ liệu.

Để thực hiện công việc này cần phải đưa hàm $y(x_{1,i}, x_{2,i},...)$ vào phương trình tính sai số dư, và sau đó lấy đạo hàm phương trình đó đối với một trong các hệ số bằng không. Điều này dẫn đến kết quả là một phương trình cho mỗi hệ số dưới dạng các hệ số và các điểm dữ liệu khác, mà sau đó giải các phương trình với những hệ số này. Các hàm số hồi quy cài sẵn của Excel chú ý đến tất cả các phép tính hồi quy tuyến tính bội cho bạn.
Cùng với các hệ số của phương trình hồi quy, Excel còn tính một số dữ liệu thống kê về thích hợp đường cong như sau:

- Sai số tiêu chuẩn của ước lượng y (S_y, x) .
- Chỉ số tương quan (hệ số xác định) (r^2) .
- Sai số tiêu chuẩn của các hệ số $(S_A, S_B,...)$.
- Số liệu thống kê F.
- Sai số bậc tự do.
- Tổng bình phương của phép hồi quy và của các số dư.

6.1.1.1. Sai số chuẩn của ước lượng y

Sai số chuẩn của ước lượng y là sự ước tính về sai số trong đơn trị y đã được tính bằng phương trình xấp xỉ. Người ta sử dụng ước tính này, cùng với phép thử t của Student, để tính các giới hạn tin cậy của đường cong tính toán. Giới hạn tin cậy là một dải gần đường cong tính toán, với mức độ tin cậy nào đó (chẳng hạn 95%), giới hạn đường cong thực. Sai số chuẩn của ước lượng y được tính bằng phương trình:

$$S_{yx} = \sqrt{\frac{\sum_{i=1}^{n} [y_i - y(x_i)]^2}{p}}$$

Ở đây p là số bậc tự do (p = n - 2 dối với đường cong tuyến tính đơn giản).

6.1.1.2. Chỉ số tương quan

Chỉ số tương quan, hay hệ số xác định, bằng bình phương của hệ số tương quan (r) và là một thước đo để tháy đường cong xấp xỉ gần với các điểm dữ liệu đến mức như thế nào. Chỉ số này có khoảng biến thiên từ 0 đến 1, với giá trị 1 cho thấy sự xấp xỉ hoàn hảo với các điểm dữ liệu. Việc làm xấp xỉ đường cong tốt sẽ có một chỉ số tương quan với giá trị lớn hơn 0,9. Chỉ số tương quan được tính bằng phương trình:

$$r^{2} = 1 - \sqrt{\frac{\sum_{i=1}^{n} [y_{i} - y(x_{i})]^{2}}{\sum_{i=1}^{n} (y_{i} - \langle y_{i} \rangle)^{2}}}$$

Ở đây:

$$\langle y_i \rangle = \frac{\sum_{i=1}^n y_i}{n}$$

là giá trị trung bình của dữ liệu y.

6.1.1.3. Sai số chuẩn của các hệ số

Sai số chuẩn của các hệ số là thước đo về sai số trong mỗi hệ số hồi quy. Sai số chuẩn trong hệ số thứ nhất (S_A) được tính bằng cách sử dụng sai số chuẩn của ước lượng y:

$$S_{A} = 1 - \sqrt{\frac{1}{n} + \frac{(x)^{2}}{\sum_{i=1}^{n} [x_{i} - (x)]^{2}}}$$
$$(x) = \frac{\sum_{i=1}^{n} x_{i}}{n}$$

Trong đó:

Tác dụng chính của sai số chuẩn của các hệ số là để thử một hệ số xem liệu qua thống kê nó có giá trị bằng 0 hay không. Vì tất cả các hệ số nhân với số hạng x tuyến tính cho nên nếu một hệ số bằng 0 thì không có tương quan giữa số hạng x đó với dữ liệu y. Để thử một hệ số, ta lấy giá trị t của Student thích hợp với khoảng tin cậy yêu cầu $(1 - \alpha)$ và các bâc tự do (p), và tính:

Nếu phương trình trên đúng, thì hệ số này là đáng kể và các giá trị của y sẽ phụ thuộc vào các giá trị của x mà nhân với hệ số này. Nếu phương trình trên sai, thì các giá trị y không phụ thuộc vào các giá trị x đó, và ta nên sử dụng giá trị 0 đối với hệ số này.

Các hệ số còn lại được xử lý theo cách tương tự. Để ôn lại chút ít về lý thuyết thống kê, xin bạn hãy tìm một cuốn sách viết về thống kê kỹ thuật. Trong sách đó cũng sẽ cung cấp cho bạn nhiều thông tin hơn về cách sử dụng các con số thống kê này như thế nào và sẽ bao cho một bảng các giá trị t của Student.

Nói chung, nếu giá trị tuyệt đối của hệ số có độ lớn lớn hơn sai số tiêu chuẩn của hệ số đó thì bạn có thể chắc chắn rằng nó là đáng kể. Nếu bạn có ít nhất 4 bậc tự do (ví dụ như 6 điểm dữ liệu cho một sự xấp xỉ tuyến tính) thì giá trị t của Student với khoảng tin cậy 95% chỉ gần bằng 2,1 và với nhiều bậc tự do hơn thì giá trị t này sẽ giảm xuống. Do vậy, một quy tắc ngón tay cái thích hợp là: nếu giá trị tuyệt đối của hệ số lớn hơn sai số tiêu chuẩn của hệ số đó 2,5 lần thì hệ số là đáng kể. Nếu nó nhỏ hơn, bạn sẽ phải tìm giá trị t đúng của Student trong sách thống kê và đưa nó vào phương trình trên để biết chắc liệu hệ số này có đáng kể không.

6.1.1.4. Số thống kê F

Người tả thường sử dụng số liệu thống kê F với một bảng các giá trị F để xác định xem dữ liệu có thực sự theo dạng đường cong không, hoặc liệu sự xấp xỉ về mặt hình thức này có phải chỉ do các biến thiên ngẫu nhiên trong dữ liệu hay không. Cũng như với các phép thử giá trị t của Student, để sử dụng số liệu thống kê F, bạn cần có một bảng

các giá trị F từ tập bảng toán học hoặc từ một cuốn sách về Toán học thống kê. Từ số bậc tự do và giới hạn tin cậy (chẳng hạn 95%), bạn lấy một giá trị F ương bảng và so sánh với giá trị F tính toán. Chừng nào mà giá trị F tính toán lớn hơn giá trị F trong bảng thì sự xấp xỉ này là do mối tương quan có thật chứ không phải do ngẫu nhiên.

Bảng giá trị F đòi hỏi 2 giá trị bậc tự do thêm vào giới hạn tin cậy. Số hạng thú nhất, n_{fl} , bằng số của các hệ số trong phương trình hồi quy trừ đi một. Số hạng thứ hai, p, là số bậc tự do tiêu chuẩn và bằng số các tập dữ liệu trừ đi số các hệ số trong phương trình đang được làm xấp xỉ với dữ liệu. Giá trị p là giá trị bậc tụ do được đưa ra bởi hàm LINEST và được sử dụng cho phép thử giá trị t của Student.

6.1.1.5. Số bậc tự do

Số bậc tự do, p, bằng số các điểm dữ liệu trừ đi số các hệ số hồi quy. Phương trình của một đường có hai hệ số: hệ số góc và hằng số hoặc số hạng chenh lệch y. Nếu bạn có mười điểm dữ liệu thì số bậc tự do sẽ là 8 (= 10 - 2). Cần có số bậc tự do cùng với nhiều bảng thống kê để tính các giới hạn tin cậy.

6.1.1.6. Tổng bình phương của phép hồi quy và các số dư

Hai số liệu thống kê tổng bình phương này là phép đo sai số vẫn còn tồn tại trong việc làm xấp xỉ đường cong. Tổng bình phương của phép hồi quy bằng tổng các sai phân giữa các giá trị dữ liệu y và mức trung bình của các giá trị dữ liệu y bình phương:

$$\sum_{i=1}^{n} [y_i - (y)]^2$$

Do vậy, nó là phép đo sự phân tán của dữ liệu ở gần mức trung bình.

Tổng bình phương của các số dư là tổng của các sai phân giữa các giá trị dữ liệu y ban đầu và các giá trị dữ liệu y được tính tương ứng trên đường cong bình phương:

$$\sum_{i=1}^{n} \left[y_i - (x_i) \right]^2$$

Do vậy, nó là phép đo sự phân tán của dữ liệu y ở gần đường hồi quy Khi bạn chia các giá trị này cho số bậc tự do, thì bạn sẽ nhận được phương sai của dữ liệu ở gần mức trung bình và phương sai lân cận đường hồi quy. Lấy căn bậc hai của phương sai, và bạn sẽ có được độ lệch chuẩn của dữ liệu ở gần mức trung bình và độ lệch chuẩn của dữ liệu lân cận đường hồi quy.

6.1.2. Các phép tính hồi quy tuyến tính

6.1.2.1. Phép hồi quy với các hàm bảng tính

Trong Excel chúng ta thực hiện phép giải tích hồi quy tuyến tính bằng các hàm LINEST, LOGEST, TREND và GROWTH. Hàm LINEST thực hiện phép hồi quy tuyến tính đơn giản trên một tập điểm dữ liệu. LOGEST là biến thể của phép hồi quy tuyến tính mà làm xấp xỉ phương trình sau với dữ liệu:

 $y = A(B^{x1}) (C^{x2})...$

Các hàm LINEST và LOGEST khai báo các hệ số của công thức. Hàm TREND và GROWTH khai báo đường cong được thích ứng với dữ liệu. Tất cả 4 hàm này đều khai báo các mảng dữ liệu chứ không phải là các giá trị đơn. Bạn phải đưa 4 hàm này vào trong nhóm các ô, hoặc sử dụng hàm INDEX để rút một phần tử đơn từ mảng dữ liệu đó. Chương 1 mô tả cách đưa một hàm vào trong khối ô. Khi bạn đưa một hàm mảng vào nhóm các ô, bạn không thể thay đổi bất cứ một ô riêng lẻ nào trong nhóm đó. Bạn phải thay đổi toàn bộ nbóm, hoặc phải xoá toàn bộ nhóm và sau đó mới thực hiện những thay đổi.

Hàm LINEST và LOGEST có cú pháp sau:

LINEST(y-array, x array, const, statistics) LOGEST(y-array, x array, const, statistics)

Ở đây y-array nói đến các điểm dữ liệu y, x-array nói đến một hay nhiều tập điểm dữ liệu x, const là giá trị lôgic kiểm tra số hạng không đổi, và statistics là giá trị lôgic xác định liệu có đưa ra các giá trị thống kê hay không.

Nếu số hạng x-array bị loại bỏ thì sẽ sử dụng tập hợp các số {1, 2, 3,...}. Nếu số hạng const là TRUE hoặc bị loại bỏ thì số hạng không đổi trong việc vẽ đường cong (A) được tính bình thường. Nếu const là FALSE thì số hạng không đổi này buộc phải bằng 0 đối với LINEST hoặc bằng 1 đối với LOGEST. Nếu số hạng statistics đúng thì một bảng gồm tám giá trị thống kê trở lên được đưa ra cùng với các hệ số của phương trình.

6.1.3. Các phép tính hồi quy đa thức

Mặc dù các hàm hồi quy dữ liệu sẵn có của Excel rõ ràng là đã không được cài đặt để thực hiện phép hồi quy đa thức, nhưng bạn vẫn có thể dễ đàng thực hiện công việc này. Phép hồi quy đa thức làm xấp xỉ dữ liệu bằng một đường có dạng:

$$y = A + Bx + Cx^2 + \dots$$

Có thể làm xấp xỉ phương trình này với phương trình đệ quy đa tuyến tính bằng cách cho:

$$\mathbf{x}_{1,i} = \mathbf{x}_i$$
$$\mathbf{x}_{2,i} = \mathbf{x}_i^2$$
$$\mathbf{x}_{3,i} = \mathbf{x}_i^3$$

Làm xấp xỉ dữ liệu về độ dẫn nhiệt một lần nữa, nhưng lần này sử dụng phương trình đa thức bậc ba (lên tới x^3).

Hãy lần lượt thực hiện các thao tác sau đây :

- Sử dụng bảng tính ở ví dụ trước. Hãy lưu nó với một tên khác nếu bạn muốn giữ nó.
- Gõ dòng He so nhiet cua GaAs; Lap hoi quy da thuc trong ô A1.
- Chọn cột C và chọn lệnh Delete trên bảng chọn Edit.
- Chọn cột B và C, và chọn lệnh Insert trên bảng chọn Edit.
- Gõ T² vào ô B3 và T³ trong ô C3.
- Trong ô B4, $g\tilde{o} = A4^2$ và sao chép nó sang các ô B5:B16.
- Trong ô C4, gõ = $A4^3$ và sao chép nó sang các ô C5:C16.
- Định dạng các ô B4:C16 là Scientific với 2 chữ số thập phân.

- Bây giờ chúng ta đưa vào phương trình ước lượng mới cho K. Chúng ta dùng ký hiệu chữ C có dấu gạch dưới (C_) để thay cho ký hiệu của hệ số C vì chữ C là một từ riêng của Excel. Tất cả các ô sẽ cho thấy giá trị sai #REF! do chưa định tên cho các hệ số này.

- Trong ô E4, gõ công thức = $A+B*A4+C_*B4+D*C4$ và sao chép dòng này sang các ô E5:E16.

- Bây giờ bạn hãy đi chuyển bảng này ra xa để tạo khoảng trống để hiển thị các kết quả hồi quy, và sau đó mở rộng bảng hồi quy.

- Chọn A3:E16 và kéo ô này theo đường viền của nó đến A14:E27.

- Chọn G5:J12 và kéo nó đến B4:E11.

- Chọn E5:E11 và kéo nó đến G5:G11.

- Mở rộng các cột F và G lần lượt tới 7 và 25 ký tự.

- Gõ D trong ô C5, và C trong ô D5, B trong ô E5, và A trong ô F5.

- Vẽ đường viền cho các ô E5, F5, E11:F11, F11, và G6:G10.

- Chọn các ô D6:D10 và chọn lệnh *Border* trên bảng chọn *Format*. Nhấn chuột lên hộp *Right* để bỏ chế độ kiểm tra, và sau đó nhấn chuột vào Ok.

Chọn các ô C5:F6 và chọn lệnh Insert / Name / Create. Hãy đặt tên cho ô D6 là C_
chứ không phải là C.

- Bây giờ chúng ta có thể thực hiện phép hồi quy.

- Chọn các ô C6:F10 và gõ công thức:

= LINEST(D15:D27, A15:C27, TRUE, TRUE)

- Ân Ctrl-Shift-Enter.

Bảng tính bây giờ sẽ giống như trong hình 6.5. Giá trị của $r^2 = 0,9989$ chứng tỏ rằng đường cong này rất xấp xỉ với dữ liệu. Chúng ta sẽ vẽ đồ thị để thấy rõ hơn điều đó như ở hình 6.6.

| 17.5 | A | В | C | D | Έ | F | C |
|------|--------|-------------|---------------|-------------|------------|-------------|----------------------------|
| 1 | Độ dẫn | nhiệt của đ | iAAs, Lập đườ | 19 cong xấp | xi da thre | | |
| 2 | | 1 | | | | | |
| 3 | | | * | | | | |
| 4 | | Báng kối q | Ty | | | | |
| 5 | | | D | С | В | A | |
| 6 | | | -2.27273E-09 | 4.834E-06 | -0.00362 | 1.0778511 | Các hệ số |
| 7 | | | 2.17043E-10 | 3.598E-07 | 0.000187 | 0.0303105 | Saisé Std của các hệ số |
| 8 | | r^2 | 0.998970639 | 0.0038932 | #N/A | #N/A | Sai số Std của ước lượng Y |
| 9 | 1 | F | 2911.430115 | 9 | #N/A | #N/A | Bậc tự do |
| 10 | | Tổng b.p | 0.132384511 | 0.0001364 | #N/A | #N/A | |
| 11 | | | Hồi quy | Residual | | | |
| 12 | | | | | | | |
| 13 | | | | | | | |
| 14 | T (K) | T*2 | T^3 | K(W/cm·K) | KI | Ghi chú : K | 1 = ước lượng của K |
| 15 | 250 | 6.25E+04 | 1.56E+07 | 0.445 | 0.440 | | |
| 16 | 300 | 9.00E+04 | 2.70E+07 | 0.362 | 0.366 | | |
| 17 | 350 | 1.23E+05 | 4.29E+07 | 0.302 | 0.306 | | |

Hinh 6.5: Độ dẫn nhiệt của GaAs: vẽ đường cong bằng phép hồi quy đa thức.



Hình 6.6: Làm xấp xỉ đường cong đã vẽ với độ dẫn nhiệt của GaAs.

Bạn có thể thử dùng đường cong bậc cao hơn để có được sự xấp xỉ tốt hơn với dữ liệu, nhưng với phép hồi quy đa thức thì đường cong này thường sẽ dao động nếu bậc quá cao. Trong trường hợp này, mặc dù đường cong trùng khớp với tất cả các điểm dữ liệu, nhưng nó thường không phải là một công cụ dự báo chính xác đối với các điểm nằm trong các điểm dữ liệu. Bạn nên thường xuyên vẽ đồ thị các kết quả của phép hồi quy để đảm bảo rằng đường cong mà bạn đã tính chính là cái bạn mong muốn.

6.1.4. Kiểm tra các số liệu thống kê

Trong thực tế các kỹ sư luôn phải kiểm tra sự tương ứng của các số liệu thống kê với đường cong xấp xỉ mà họ vừa mới tìm ra theo cách đã trình bày ở trên. Hãy xét tiếp ví dụ trước. Đầu tiên phải kiểm tra các hệ số của phép hồi quy (A, B, C và D) với khoảng tin cậy 95%:

$$\alpha = (1 - 0.95) = 0.05$$

 $\alpha/2 = 0,025$ p = 9 (rút ra từ bảng tính EXCEL) ta/2, p = t0.025,9 = TINV(0.05,9) = 2.262 (từ bảng các giá trị của Student) $S_A = 0,0303$ (rút ra từ bảng tính) $T0.025,9S_A = 2,262 \times 0,0303 = 0,0685$

Giá trị này nhỏ hơn nhiều so với A, nên A là số đáng kể. Áp dụng cách phân tích tương tự đối với các hệ số khác.

Sau đó, kiểm tra xem dữ liệu này có thực sự tương quan với đường cong không hoặc liệu mối tương quan về mặt hình thức này có phải là kết quả của một quá trình ngẫu nhiên không.

Giá trị F trong bảng các giá trị F với p = 9 và $n_i = 3$ và khoảng tin cậy 0,95 là FINV(0.05,3,9) = 3,86, tức là nhỏ hơn giá trị F = 2,911 đã tính, vì vậy mối tương quan giữa dữ liệu và đường cong là đáng kể và không phải là sự kiện ngẫu nhiên.

6.2. XỬ LÝ CÁC HÀM PHỨC

Nếu một hàm không thể được tuyến tính hoá thành dạng yêu cầu cho phép hồi quy tuyến tính thì bạn không thể sử dụng các hàm có sẵn (buil-in function) của Excel để xác định một phương trình xấp xỉ tốt nhất với dữ liệu. Ví dụ, một hàm mũ ($y = Ae^{BX}$) có thể được tuyến tính hoá bằng cách lấy lôgarit tự nhiên của nó, ln(y) = ln(A) + Bx, và chấp nhận ln(A) và B là các hệ số hồi quy hơn là A và B của phương trình ban đầu.

Mặt khác một hàm mũ kép như y = $Ae^{Bx} + Ce^{Dx}$ không thể được tuyến tính hoá trừ khi bạn biết B và D.

Như đã đề cập ở phần trước của chương này, mặc dù các hệ số đã tính toán cho phương trình được tuyến tính hoá là xấp xỉ tốt nhất (bình phương nhỏ nhất) với phương trình đó, nhưng chúng thường không phải là thích hợp tốt nhất đối với hàm ban đầu. Ví dụ, phép hồi quy dựa vào kiểu đã được tuyến tính hoá của một phương trình mũ đơn ở trên sẽ tính các giá trị tốt nhất của ln(A) và B đối với phương trình lôgarit. Chúng sẽ không cần phải là các giá trị tốt nhất của A và B đối với phương trình mũ, nhưng chúng sẽ phải lân cận với các giá trị tốt nhất đó.

6.2.1. Hiệu chỉnh bằng tay

Để làm xấp xỉ các phương trình mà không thể thực hiện bằng phép phân tích hồi quy, bạn phải hiệu chỉnh liên tiếp các hệ số để tìm giá trị cực đại của r², tương ứng với số tối thiểu trong sai số bình phương số dư.

Hãy tăng hoặc giảm một hệ số cho đến khi bạn tìm thấy giá trị cực đại của r². Sau đó, hiệu chính hệ số tiếp sau để tìm giá trị cực đại, và cứ như vậy cho đến khi bạn đã hiệu

chỉnh tất cả các hệ số. Sau đó lại bắt đầu với hệ số đầu tiên để xem việc thay đổi bất kỳ hệ số nào trong các hệ số khác còn lại có làm thay đổi vị trí của giá trị cực đại đối với hệ số này không. Tiếp tục hiệu chỉnh các hệ số đến khi bạn thấy các giá trị mà đồng thời cho giá trị cực đại của hệ số tương quan, lúc đó công việc của bạn đã hoàn thành.

Đôi khi, việc thay đổi hệ số đầu tiên làm thay đổi giá trị cực đại của một hệ số khác, nhưng ngược lại việc thay đổi hệ số khác sẽ làm thay đổi giá trị cực đại của hệ số đầu tiên. Thuật toán hiệu chỉnh của bạn lúc đầu dao động qua lại giữa hai hệ số này và không bao giờ hội tụ. Hãy cố gắng hiệu chỉnh các giá trị này với một lượng nhỏ hơn và xem liệu nó sẽ hội tụ hay không. Nếu nó không hội tụ, thì bạn phải biết vận dụng óc suy xét và kiến thức khoa học của mình về phương trình đang được làm xấp xỉ để xác định các giá trị tốt nhất. Hãy nhớ rằng có thể có hai cực trị.

6.2.2. Hiệu chỉnh tự động

Trong ví dụ trước, chúng ta đã hiệu chỉnh bằng tay các hệ số của phương trình đang làm xấp xỉ với các dữ liệu cho đến khi r^2 tăng tới mức tối đa. Đôi khi, chúng ta cần những sự hiệu chỉnh bằng tay khi phương trình phi tuyến hoàn toàn hoặc khi phương trình tạo ra các giá trị lạ lân cận nghiệm. Các phương trình có diểm cực trị cục bộ thuộc loại phương trình này. Chúng ta cần tạo ra những thay đổi qua trực giác để tìm nghiệm đúng hoặc nghiệm chính xác nhất. Tuy nhiên, trong hầu hết các trường hợp, có thể áp dụng sự tiếp cận cơ học để tìm nghiệm.

Từ Excel 4.0 đến Excel 97 đã bao gồm chương trình Solver bổ sung, mà chúng ta có thể sử dụng để làm tăng tối đa r² một cách tự động và tìm sự xấp xỉ tốt nhất đối với một tập dữ liệu. Solver được thiết kế để điều chỉnh nhanh nhậy một số giá trị đến khi thoả mãn một tiêu chuẩn nào đó. Nếu đã có một phiên bản Excel cũ trước đó, bạn có thể viết Macro để thực hiện công việc tương tự.

6.2.2.1. Sử dụng chương trình Solver

Lập bảng tính cho việc sử dụng *Solver* giống như việc lập bảng tính trong ví dụ trước. Việc bổ sung *Solver* thay thế bạn một cách hữu hiệu, đưa vào các giá trị của A và B theo dõi xem r² thay đổi như thế nào. Ngoài ra, có thể sử dụng các công thức chuỗi để tính sai số tiêu chuẩn của hệ số y và r². Bạn có thể đã sử dụng các công thức chuỗi này trong ví dụ trước, nhưng chúng không dễ hiểu. Trước hết, hãy thay thế các công thức trong các cột bằng các công thức chuỗi. Sau đây là trình tự các thao tác:

1. Bắt đầu với bảng tính từ ví dụ trước.

2. Chọn các cột D và E và xoá bỏ các nội dung của chúng bằng lệnh *Clear* trên bảng chọn *Edit*.

3. Trong ô H8, gõ công thức:

=SQRT(SUM((B4:B16-C4:C16)^2/FREE)

- 4. Ân Ctrl-Shift-Enter
- 5. Trong ô H9, gõ công thức:

```
=1-SUM((B4:B16-C4:C16)<sup>2</sup>)/SUM((B4:B16-AVEY)<sup>2</sup>)
```

6. Ấn Ctrl-Shift-Enter.

7. Đặt tên ô H9 là rsq.

Hai công thức mảng này thay thế hoàn toàn cho hai cột tính ở kiểu trước của bảng tính này. Bằng cách thay thế các tham chiếu ô trong công thức bằng các tham chiếu mảng và việc nhập công thức vào bằng cách ấn *Ctrl-Shift*. chúng ta đã nén hai cột tính vào hai ô.

Trong một công thức mảng, Excel làm thích ứng ô đầu tiên trong mỗi tham chiếu mảng và tính kết quả. Sau đó, nó thực hiện điều này đối với tập giá trị tiếp theo, và cứ như vậy, cho đến khi nó đạt tới đầu cuối của tham chiếu mảng. Ví dụ, trong ô H8, Excel lấy các giá trị trong ô B4:B22 trừ đi các giá trị trong ô C4:C22. Kết quả là một danh sách gồm 19 số được chuyển đến hàm SUM. Hàm SUM cộng thêm danh sách này, kết quả được chia cho giá trị FREE, và sau đó lấy căn bậc hai. Nếu như bạn đã ấn *Enter* thay vì ấn *Ctrl-Shift-Enter*, thì công thức này sẽ chỉ áp dụng với các ô B8 và C8 (các phần tử của mảng cùng hàng với công thức).

Bây giờ chúng ta hãy sử dụng Solver để tìm nghiệm.

- 8. Đặt giá trị của A và B trong các ô G3 và G4 là 0,5 và 500.
- 9. Chọn lệnh Solver trên bảng chọn Formula. Hộp thoại Solver Parameters xuất hiện.

10. Trong hộp thoại này, gõ rsq trong hộp Set Cell, nhấn chuột vào nút Max đối với lựa chọn Equal to, gõ A, B trong hộp By Changing Cell. Hộp thoại của bạn sẽ giống hình 6.9.

| olver Parameters | | <u> </u> |
|-----------------------------|------------------------|---------------------|
| Set Target Cell: rsq | N. | <u>S</u> olve |
| aqual To: 💽 Max 🦵 Min | ✓ Value of: 0 | Close |
| By Changing Cells: | | |
| A,B | <u>S</u> <u>G</u> uess | |
| Subject to the Constraints: | | Options |
| | <u>A</u> dd | |
| | Change | a state and a state |
| | | Reset All |
| | |] Help |

Hình 6.9: Sử dụng Solver với bảng tính cho mặt cắt ngang của sự ion hoá electron.

Bạn đã tạo lập Solver để thử làm tăng tối đa giá trị của rsq bằng cách hiệu chỉnh các giá trị của A và B. Bây giờ hãy chạy Solver để tìm nghiệm.

11. Nhấn chuột vào nút Solver và đọi Solver tìm nghiệm.

Trong khi Solver đang làm việc, nó hiển thị giá trị của rsq ở đáy màn hình vì vậy bạn có thể nhìn thấy điều gì đang diễn ra. Khi Solver tìm thấy nghiệm, nó hiển thị một hộp thoại thể hiện ở hình 6.10. Bạn có thể chọn để giữ nghiệm này hoặc khôi phục các giá trị ban đầu. Bạn cũng có thể in một trong số các báo biểu đã liệt kê trong hộp Reports. Nếu đang sử dụng Scenario Manager, bạn có thể cất giữ nghiệm này dưới dạng một kịch bản (Scenario) bằng nút Save Scenario.

12. Nhấn OK trong hộp thoại Solver.

Kết quả cuối cùng, như thể hiện trong hình 6.11. Kết quả này hơi khác so với kết quả đã tìm thấy theo phương pháp hiệu chỉnh bằng tay, vì chúng ta đã dừng lại ở độ chính xác đến hàng nghìn với phương pháp bằng tay. Số tôi đa cục bộ nào đó mà có thể làm bạn thích thú (và *Solver*) cũng ở trong vùng nghiệm này. Sau khi *Solver* đã tìm nghiệm xong, bạn có thể phải khảo sát vùng gân nghiệm bằng cách thay đổi A và B để xem liệu có một nghiệm nào khác có giá trị r² lớn hơn không.



Hình 6.10: Hộp thoại Solver sau khi tìm thấy nghiệm.

| 83 | A | В | C | D | E | F | G | H |
|----|---------|-----------|------------|------------|------------------------|-----|--------------|-------------|
| 1 | Mặt cát | nyang | đối với sự | ion hoś da |) các elect | ron | trong Heliva | 0 |
| 2 | 1 | | | | | | | |
| 3 | E(eV) | $S(pa^2)$ | UL của S | | | A | 0.44365 | |
| 4 | 150 | 0.419 | 0.419 | | | В | 432.70156 | |
| 5 | 175 | 0,408 | 0.406 | | 1 | | | |
| 6 | 200 | 0.394 | 0.393 | 1.1 | Trung binh cua y < y> | | | 0.303 |
| 7 | 250 | 0.365 | 0.365 | | Độ tự đo | | | 11.000 |
| 8 | 300 | 0.337 | 0.339 | | Sai số của ước lương Y | | | 0.002230368 |
| 9 | 350 | 0.313 | 0.315 | | r^2 | | | 1.00E+00 |
| 10 | 400 | 0.292 | 0 293 | | | | | |
| 11 | 450 | 0.272 | 0.274 | | | | | |
| 12 | 500 | 0.255 | 0.257 | | | | | |
| 13 | 550 | 0.240 | 0.242 | | | | | |
| 14 | 600 | 0.227 | 0.228 | | 1.1 | | | |

Hình 6.11: Bảng tính sau khi Solver đã điều chỉnh các hệ số.

6.2.2.2. Sử dụng thủ tục Visual Basic để điều chỉnh tự động

Nếu vì lý do nào đó mà *Solver* không hoạt động như mong muốn, chúng ta có thể viết một thủ tục VBA để điều chỉnh các hệ số và tìm lời giải. Sau đây là thủ tục *DieuChinh*, nó sẽ điều chỉnh A và B tăng lên hoặc giảm xuống để sao cho giá trị của r² tăng lên.

Option Explicit ' DieuChinh Macro 'Thu tuc tim duong cong xap xi Sub DieuChinh Dim db1A As Double Dim db1B As Double Dim db1DA As Double 'Delta A. Dim db1DB As Double 'Delta A Dim dblStop As Double 'Stop value Dim fTest As Boolean ' Change flag Dim dblRsq As Double ' The value or r squared Dim dblRsqMax As Double ' The largest value or r squared found Initialize the variables. dblA = ActiveSheet.Range("A").Value dblB = ActiveSheet.Range("B").Value dblDA = dblA / 10dbIDB = dbIB / 10dlbStop = dlbA / 10000dlbRsqMax = ActiveSheet.Range('rsq").Value 'Look until youreach the stopping value. Do While (dblDA > dblStop) 'fTest is True if A or B have changed in an iteration. fTest = TrueDo While (fTest = True) fTest = False

'Increase A and update the WorkSheet, see if r^2 increase.

dblA = dblA + dblDA

```
ActiveSheet.Range("A").Value = dblA
```

Calculate

dblRsq = ActiveSheet.Range("rsq").Value

If (dblRsq > dblRsqMax) then

'If r^2 increases, update r^2max and set the flag.

dblsqMax = dblRsq

fTest = True

Else

'If r² decreases, try decreasing A and check again.

dblA = dblA - 2 * dblDA

ActiveSheet.Range("A").Value = dblA

Calculate

dblRsq = ActiveSheet.Range("rsq").Value

If (dblRsq > dblRsqMax) Then

'If r² increases, update r²max and set the flag.

dblRsqMax = dblRsq

fTest = True

Else

'If r² decreases, reset A to its original value.

dblA = dblA + dblDA

ActiveSheet.Range("A").Value = dblA

Calculate

End If

End If

'Now do exactly the same for B.

'Increase B and update the worksheet, see if r² increases.

dblB = dblB + dblDB

ActiveSheet.Range("B").Value = dblB

Calculate

dblRsq = ActiveSheet.Range("rsq").Value

If (dblRsq > dblRsqMax) then

'If r^2 increases, update r^2max and set the flag dblRsqMax = dbl Rsq fTest = True

Else

'If r² decreases, try decreasing B and check again. dblB = dblB - 2 * dblDBActiveSheet.Range("B").Value = dblB Calculate dblRsq = ActiveSheet.Range("rsq").Value If (dblRsq > dblRsqMax) Then 'lf r² increases, update r²max and set the flag. dblRsqMax = dblRsqfTest = TrueElse 'If r² decreases, reset B to its original value. dblB = dblB + dblDBActiveSheet.Range("B").Value = dblB Calculate End If End If Loop 'Get here if neither A or B are changed during an iteration. 'Decrease both Delta A and Delta B by a factor of 10. dbIDA = dbIDA / 10

dblDB = dblDB / 10

Loop

End Sub

Thủ tục nói trên sẽ hoạt động như sau: đầu tiên nó sao chép các trị số ban đầu của A và B từ bảng tính và khởi động các giá trị ban đầu nào đó, tiếp theo là vòng tặp tiếp tục mãi cho đến khi giá trị của Delta A (tức là dblDA) nhỏ hơn trị số dừng. Vòng lặp này xác định số chứ số thập phân sau dấu phẩy của nghiệm. Tiếp theo, cờ hiệu (tức là fTest) bị xoá và vòng lặp đã bắt đầu sẽ tiếp tục cho đến lúc fTest = True. Cờ hiệu (fTest) được đặt bằng False ở lúc bắt đầu của mỗi lần xấp xỉ của vòng lặp, và được đối thành True nếu có bất cứ bệ số nào thay đổi. Nếu không có hệ số nào thay đổi thì tức là cực đại của r^2 đã được tìm thấy và vòng lặp sẽ kết thúc. Bên trong vòng lặp, khối đầu tiên của các dòng lệnh sẽ làm tăng giá trị của A bởi DA, cải thiện trang tính, và kiểm tra xem liệu r^2 có tăng hay không. Nếu có tăng thì Macro sẽ cất giữ giá trị của r^2 đặt cờ hiệu fTest, nếu ngược lại thì macro sẽ làm giảm A bởi 2*dblDA và r^2 lại bị kiển tra lại lần nữa... Nếu bây giờ nó tăng lên thì r^2 sẽ được lưu giữ lại và cờ hiệu fTest được đặt. Nếu r^2 không tăng lên thì A vẫn giữ lại giá trị của mình.

Khối dòng lệnh tiếp theo cũng hoạt động tương tự nhưng đối với B chứ không phải đối với A. Vòng lặp này tiếp tục mãi đến khi mà r² không thể tăng hơn nữa nhờ sự thay đổi giá trị của A và B. Tiếp theo Delta A và Delta B (tức là dblDA và dblDB) sẽ bị chúng ta giảm đi 10 lần, Delta A được kiểm tra xem liệu nó đã đạt đến giá trị chuẩn dừng chưa, và nếu chưa thì vòng lặp lại bắt đầu lần nữa. Nếu đã dạt đến chuẩn dừng thì thôi, thủ tục này kết thúc.

Sau đây chúng ta sẽ áp dụng thủ tục nói trên cho bài toán ví dụ cũ. Hãy thực hiện lần lượt các thao tác sau:

- 1. Bắt đầu với bản sao của trang tính cũ.
- 2. Chọn lệnh Insert/ Macro/ Module để tạo ra trang tính module mới
- 3. Đặt tên trang tính module là DieuChinh.
- 4. Nhập thủ tục DieuChinh vào trang tính module.

5. Mở bảng tính như hình 6.12, nhấp chuột vào núm *Drawing*, chọn núm *Create* Button trên thanh công cụ *Drawing* và kéo núm trên bảng tính. Trong hộp thoại Asign Macro vừa xuất hiện, hãy chọn macro **DieuChinh** và nhấp vào OK.

| 義前 | A | B | 0 | D | E | F | G- | 治理·Hand |
|----|---------|---------|------------|---------|------------|--------------------------|--------------|-------------|
| 1 | Mặt cát | ngang | đối với sự | ion ho: | á do các e | lectron | trong Helium | 1 |
| 2 | | | 1 | | | | | |
| *3 | E(eV) | S(pa^2) | UL của S | | | A | 0.4437 | |
| 4 | 150 | 0.419 | 0.419 | | | В | 432.6 | |
| 5 | 175 | 0.408 | 0.406 | | | | | |
| б | 200 | 0.394 | 0.393 | | | Trung binh cua y <y></y> | | 0.303 |
| 7 | 250 | 0.365 | 0.365 | | | Độ tư do | | 11.000 |
| 8 | 300 | 0.337 | 0.339 | | | Sai số của ước lượng Y | | 0.002235607 |
| 9 | 350 | 0 313 | 0.315 | | | r^2 | | 1.00E+00 |
| 10 | 400 | 0.292 | 0.293 | | | | 1 | |
| 11 | 450 | 0.272 | 0.274 | | 1 | | | |
| 12 | 500 | 0 255 | 0.257 | | | | | |
| 13 | 550 | 0.240 | 0.242 | | | | | |
| 14 | 600 | 0.227 | 0.228 | | C | | | |
| 15 | 650 | 0.216 | 0.216 | | | | | |
| 16 | 700 | 0.205 | 0.205 | | | | | |

Hình 6.12: Xấp xỉ phi tuyến nhờ "Visual Basic fof Application".

- 6. Đổi tên trên núm thành XapXi SL.
- 7. Đóng hộp công cụ Drawing.

8. Thay đổi các giá trị của A trong ô G3 thành 0.5 và của B trong ô G4 thành 500.

9. Nhấp vào núm XapXi SL.

Các giá trị của A và B sẽ tiếp tục được điều chỉnh cho đến khi mà tìm được nghiệm như đã thể hiện trên hình 6.12.

6.3. TRA BẢNG VÀ PHÉP NỘI SUY

Nhiều khi chúng ta có các dữ liệu mà không thể làm xấp xỉ chúng với bất cứ phương trình đơn giản hay phương trình tương đối phức tạp nào. Trong trường hợp đó, cách tốt nhất là sử dụng một bảng dữ liệu và một hàm tìm kiếm trong bảng. Một hàm tìm kiếm trong bảng sẽ tìm và nội suy các giá trị trong bảng với giá trị x đặc biệt. Như thế có nghĩa là bạn sẽ vẽ một đường cong đơn với một vài điểm dữ liệu trong vùng lân cận của điểm mà bạn quan tâm chứ không phải là vẽ một đường cong phức tạp phù hợp với toàn bộ tập dữ liệu.

Việc tìm kiếm trong bảng được thực hiện bằng các hàm HLOOKUP, và VLOOKUP, MATCH, và FASTMATCH. Các hàm HLOOKUP và VLOOKUP tìm kiếm một giá trị trong một cột của bảng và trả về giá trị trong một cột khác trên cùng một hàng. Các hàm MATCH và FASTMATCH tìm kiếm một giá trị trong một bảng và trả về vị trí của ô mà chứa giá trị đó.

Phương pháp nội suy có thể được mã hoá dưới dạng một Macro hoặc bằng các công thức ô. Sự khác biệt trong các phương pháp nội suy khác nhau là ở phương trình được sử dụng để ước tính giá trị của hàm giữa hai điểm dữ liệu đã biết. Đơn giản nhất và phổ biến nhất là phép nội suy tuyến tính. Thực ra, phương pháp đơn giản nhất là sử dụng các hàm tìm kiếm trong bảng và chấp nhận giá trị trả về hơn là nội suy giữa các giá trị trong bảng. Trong nhiều trường hợp, phương pháp này có thể đáp ứng và làm giúp bạn rất nhiều việc. Hơn nữa, ta thường gặp các đường cong bậc hai và bậc ba, mà đều xấp xỉ với ba hoặc bốn điểm dữ liệu. Các hàm phức tạp hơn là hàm Splin và đa thức Trêbưsép. Bạn nên tham khảo một vài cuốn sách khác về giải tích số ứng dụng để biết thêm thông tin về các hàm này.

6.3.1. Phép nội suy tuyến tính

Phép nội suy tuyến tính bao gồm việc nối đơn giản hai điểm dữ liệu ở cả hai phía của giá trị đang được nội suy với một đường thẳng. Nếu các điểm dữ liệu này không cách xa về một phía thì phép nội suy tuyến tính được thực hiện rất tốt. Nó cũng thực hiện đơn giản hơn so với các phép nội suy bậc cao hơn.

Công thức nội suy ở dạng Lagrăng đối với phép nội suy tuyến tính là:

$$y = \frac{(x - x_2)}{(x_1 - x_2)} y_1 + \frac{(x - x_1)}{(x_2 - x_1)} y_2$$

Ở đây x_1 , x_2 và y_1 , y_2 là các điểm dữ liệu trong bảng và x (giá trị đang được nội suy) nằm giữa x_1 và x_2 .

6.3.2. Phép nội suy bậc ba (nội suy lập phương)

Phép nội suy bậc ba chính xác hơn phép nội suy bậc hai, không những vì nó sử dụng đường cong bậc cao hơn mà còn quan trọng hơn là bởi vì nó đối xứng hơn về phạm vi đang được nội suy. Phép nội suy bậc ba làm xấp xỉ đường cong bậc ba với bốn điểm dữ liệu trong một hàng, với phép nội suy giữa trung tâm hai điểm. Phương trình nội suy, ở dạng Lagrăng là:

$$y = \frac{(x - x_2)(x - x_3)(x - x_4)}{(x_1 - x_2)(x_1 - x_3)(x_1 - x_4)}y_1 + \frac{(x - x_2)(x - x_3)(x - x_4)}{(x_2 - x_1)(x_2 - x_3)(x_2 - x_4)}y_2$$
$$y = \frac{(x - x_1)(x - x_2)(x - x_4)}{(x_3 - x_1)(x_3 - x_2)(x_3 - x_4)}y_3 + \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_4 - x_1)(x_4 - x_2)(x_4 - x_3)}y_4$$

Trong đó x_1 , x_2 , x_3 , x_4 , y_1 , y_2 , y_3 , và y_4 liên tiếp là các giá trị x và y từ bảng. Giá trị đang được nội suy (x) nên nằm giữa x_2 và x_3 để nhận giá trị tốt nhất cho y.

Để có thể so sánh phép nội suy bậc ba với phép nội suy tuyến tính, hãy đưa bảng nội suy bậc ba vào cùng bảng tính với phép nội suy tuyến tính. Vì công thức nội suy bậc ba phức tạp hơn công thức tuyến tính, cho nên bạn hãy tạo một Macro để tính toán thay vì đưa nó vào bảng tính. Ngoài ra, công thức nội suy bậc ba có thể được thực hiện với một vòng lặp ngắn FOR/NEXT trong một Macro. Đầu tiên bạn lập một bảng nội suy với các giá trị thử nghiệm giống với bảng nội suy tuyến tính.