

LÊ VĂN DOANH    TRẦN KHẮC TUẤN    LÊ ĐÌNH ANH

**101 THUẬT TOÁN VÀ CHƯƠNG TRÌNH  
BÀI TOÁN KHOA HỌC KỸ THUẬT VÀ KINH TẾ  
BẰNG NGÔN NGỮ C**

*In lần thứ 6, có chỉnh sửa*

EBOOKBKMT.COM  
Tài liệu kỹ thuật miễn phí



NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT  
HÀ NỘI – 2006

## **Lời giới thiệu**

Khác với PASCAL, là ngôn ngữ cấp cao được xây dựng với mục đích để học tập và giảng dạy lập trình, ngôn ngữ C và ngôn ngữ phát triển của nó là C++ được xây dựng xuất phát từ những yêu cầu nội tại của tin học, vì thế C phản ánh những thành tựu mới nhất của phần mềm công nghệ tin học đó là tính cỗ động và rất gọn khi lập trình, có thể thâm nhập vào mọi nguồn dữ liệu và hệ thống. C có nhiều mô hình tổ chức bộ nhớ và nói chung người lập trình có thể biết sự tiến triển của chương trình trong bộ nhớ. C có nhiều toán tử rất mạnh. Chương trình viết bằng ngôn ngữ C có thể modun hóa và ghép nối dễ dàng. C có tất cả những ưu điểm của ngôn ngữ cấp cao nhưng cũng có những đặc trưng của hợp ngữ...

Vì những ưu điểm kể trên, ngày nay các chương trình lớn đều được viết bằng C hoặc đang được viết lại bằng C. Hệ điều hành WINDOWS, UNIX... đều được viết bằng C. Ngôn ngữ C trở nên bắt buộc đối với các nhà lập trình chuyên nghiệp.

Ở Việt Nam, ngôn ngữ C và C++ đang được giảng dạy trong các khoa Công nghệ tin học, Điện tử và thông tin, Đo lường, Điều khiển... của các trường đại học và chuyên nghiệp. Nhiều sách hướng dẫn lập trình ngôn ngữ C do các tác giả trong nước biên soạn và biên dịch đã được xuất bản. Tuy nhiên vẫn còn thiếu bộ sách có tính chất thực hành về C, nhất là còn thiếu những tư liệu, cẩm nang giúp đỡ cho các nhà lập trình bằng ngôn ngữ C đối với các bài toán khoa học kỹ thuật và kinh tế thường gặp.

Cuốn sách “**101 thuật toán và chương trình bài toán khoa học kỹ thuật và kinh tế bằng ngôn ngữ C**” nhằm bổ sung cho chỗ trống này. Thông qua các chương trình mẫu các tác giả mong muốn cung cấp cho bạn đọc những thuật toán và chương trình của nhiều loại bài toán khoa học kỹ thuật và kinh tế thường gặp.

Toàn bộ cuốn sách gồm 101 chương trình chia làm 11 chương.

*Chương 1* là 20 chương trình minh họa giúp bạn đọc mới bắt đầu làm quen với ngôn ngữ C. Các chương trình này có thuật toán đơn giản và được giới thiệu để minh họa cho những vấn đề cơ bản của việc lập trình ngôn ngữ C.

*Chương 2* gồm 16 chương trình dành cho các bài toán đại số với các số thực và số phức.

*Chương 3* gồm 8 chương trình đồ họa các hàm theo tham số, theo tọa độ cực, các mặt trong không gian 3 chiều.

*Chương 4* gồm 12 chương trình dành cho các bài toán vectơ, ma trận, hệ phương trình đại số tuyến tính.

*Chương 5* gồm 8 chương trình về đa thức và nội suy đa thức.

*Chương 6* gồm 4 chương trình tinh tích phân xác định và tích phân kép.

*Chương 7* gồm 3 chương trình giải các phương trình phi tuyến.

*Chương 8* gồm 3 chương trình tìm cực trị các hàm một biến và hai biến.

*Chương 9* gồm 4 chương trình giải phương trình vi phân và hệ phương trình vi phân cấp n.

*Chương 10* gồm 4 chương trình về các vấn đề xác suất và thống kê.

*Chương 11* gồm 15 chương trình xử lý văn bản, quản lý và một số bài toán ứng dụng khác.

Tất cả các chương trình đều được chạy có kết quả bằng TC91. Các chương trình của cuốn sách này đều được in trực tiếp từ máy tính.

Về bối cảnh cuốn sách này có nhiều nét giống cuốn sách “*101 thuật toán và chương trình bài toán khoa học kỹ thuật và kinh tế ngôn ngữ PASCAL*” của cùng tác giả, xuất bản năm 1993 và đã được tái bản nhiều lần. Chúng tôi cũng được sự động viên và góp ý nhận xét của nhiều bạn đọc. Nhiều vấn đề đã được đưa vào cuốn sách này. Tuy nhiên do trình độ có hạn và nhất là do có ít nguồn tư liệu nên chắc chắn cuốn sách này không tránh khỏi sai sót. Chúng tôi mong được sự chỉ dẫn, góp ý của bạn đọc. Mọi chỉ dẫn, góp ý xin gửi về Khoa Thiết bị điện Trường đại học Bách Khoa Hà Nội. Chúng tôi xin chân thành cảm ơn.

Nếu bạn đọc có yêu cầu về đĩa mềm chứa chương trình cuốn sách này xin liên hệ với khoa Thiết bị điện Trường đại học Bách Khoa Hà Nội - ĐT. 692511 chúng tôi xin cung cấp miễn phí.

### Các tác giả

## Mục lục

Lời giới thiệu	3
Mục lục	6
<b>Chương 1. Một số chương trình minh họa cơ bản lập trình ngôn ngữ C</b>	
<b>Bài số 1</b> Chương trình minh họa vào - ra	14
<b>Bài số 2</b> Chương trình minh họa cách hiển thị	16
<b>Bài số 3</b> Chương trình minh họa lệnh gán	17
<b>Bài số 4</b> Chương trình minh họa toán tử tự tăng tự giảm	19
<b>Bài số 5</b> Chương trình minh họa toán tử logic	20
<b>Bài số 6</b> Chương trình minh họa mảng và con trỏ	21
<b>Bài số 7</b> Chương trình minh họa dùng if else, phân loại học sinh theo điểm	22
<b>Bài số 8</b> Chương trình minh họa sử dụng switch, phân loại học sinh theo điểm	24
<b>Bài số 9</b> Chương trình minh họa sử dụng for.Tìm tất cả các phương án để có được 1000đ từ các tờ 500đ,200đ,100đ	25
<b>Bài số 10</b> Chương trình minh họa sử dụng while	27
<b>Bài số 11</b> Chương trình minh họa sử dụng do while, tìm tổng của cấp số tự nhiên với n cho trước	28
<b>Bài số 12</b> Chương trình minh họa khai báo và gọi hàm	29

<b>Bài số 13</b> Chương trình minh họa sử dụng hàm.	
Lập một hàm thực hiện các phép tính số học : cộng, trừ, nhân, chia của hai số thực	31
<b>Bài số 14</b> Chương trình minh họa mảng một chiều.	
Tính tổng của một dãy số	32
<b>Bài số 15</b> Chương trình minh họa mảng sử dụng con trỏ.	
Tính tổng của một dãy số	34
<b>Bài số 16</b> Chương trình minh họa mảng string. Cho một số nguyên giữa 1 và 7, tìm số ngày trong tuần tương ứng	35
<b>Bài số 17</b> Chương trình minh họa xử lý string. Đếm số chữ cái nào đó trong một dòng chữ	36
<b>Bài số 18</b> Chương trình minh họa mảng nhiều chiều.	
Tìm phần tử âm đầu tiên trong một mảng hai chiều	37
<b>Bài số 19</b> Chương trình minh họa sử dụng cấu trúc.	
Xây dựng cấu trúc nhân sự gồm họ và tên, ngày sinh, ngày tuyển dụng, bậc lương	38
<b>Bài số 20</b> Chương trình minh họa sử dụng macro	40

## Chương 2. Đại số

<b>Bài số 21</b> Tính tổ hợp chập k trong n phân tử $C_n^k$	42
<b>Bài số 22</b> Tính tích của n số thực	44
<b>Bài số 23</b> Tính tổng và tích của một dãy n số thực	45
<b>Bài số 24</b> Tìm các số dương trong một dãy số	47
<b>Bài số 25</b> Tính trung bình trong khoảng $[a,b]$ của một dãy số	49
<b>Bài số 26</b> Tìm các số của chuỗi FIBONACCI	50
<b>Bài số 27</b> Xác định số lớn nhất trong một dãy số	52

<b>Bài số 28a</b> Tìm tam giác PASCAL theo hàm nhị thức	53
<b>Bài số 28b</b> Tìm tam giác PASCAL theo biểu thức $C_{ij} = C_{i-1,j} + C_{i-1,j-1}$	55
<b>Bài số 29</b> Đảo ngược thứ tự của một dãy số	57
<b>Bài số 30</b> Sắp xếp dãy số theo chiều giảm	59
<b>Bài số 31</b> Giải phương trình trùng phương	61
<b>Bài số 32</b> Cho ba cạnh a, b, c có thể là các cạnh của một tam giác. Tìm điều kiện để có tam giác, nếu có tính chu vi, diện tích tam giác và xác định loại tam giác	63
<b>Bài số 33</b> Tính gai thừa của một số nguyên theo hàm gai thừa và gai thừa đệ quy	66
<b>Bài số 34a</b> Tìm các số nguyên tố với số n cho trước (sàng Eratosthene dynamic)	68
<b>Bài số 34b</b> Tìm các số nguyên tố giữa hai số tự nhiên cho trước (sàng Eratosthene)	71
<b>Bài số 35</b> Lập các hàm tính toán các phép tính đối với số phức : tổng, hiệu, tích, thương, khai căn, lũy thừa	72
<b>Bài số 36</b> Giải phương trình bậc hai có các hệ số phức	78
<b>Bài số 37</b> Tính tổng và tích hai số phức dùng pointer	80

### Chương 3. Đồ họa

<b>Bài số 38</b> Vẽ hàm số $y = f(x)$ . Áp dụng với $y_1 = \sin x$ , $y_2 = \frac{\sin x}{x}$	84
<b>Bài số 39</b> Vẽ đường cong theo tham số. Áp dụng với các hàm :	

Hypocycloide  $\begin{cases} x = \cos^3(t) \\ y = \sin^3(t) \end{cases}$

Trochoide  $\begin{cases} x = 2t - 3\sin(t) \\ y = 2 - 3\cos(t) \end{cases}$

và  $\begin{cases} x = (1 + \cos^2(t)).\sin(t) \\ y = \sin^2(t).\cos(t) \end{cases}$  88

<b>Bài số 40</b> Vẽ đường cong theo toạ độ cực $r = f(\theta)$ .	
Áp dụng vẽ đường xoắn ốc Archimede $r = \theta$ , $\theta \in [0,4\pi]$	
đường hoa hồng 4 cánh $r = \sin 2\theta$ , $\theta \in [0,2\pi]$	92
<b>Bài số 41</b> Vẽ quỹ đạo cycloide	96
<b>Bài số 42</b> Vẽ không gian ba chiều có khử mặt ẩn	
Áp dụng cho hàm hai biến $z = \sin x \cdot \sin y$	99
<b>Bài số 43</b> Vẽ mặt không gian ba chiều cho hàm hai biến $z = f(x,y)$	
Áp dụng cho $z = \frac{5 \sin \sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2}}$	106
<b>Bài số 44</b> Vẽ biểu đồ thống kê	111
<b>Bài số 45</b> Vẽ các quỹ đạo hành tinh của hệ mặt trời	114

#### **Chương 4. Vectơ, ma trận, hệ phương trình đại số tuyến tính**

<b>Bài số 46</b> Tính tích vô hướng và tích vectơ trong không gian ba chiều	119
<b>Bài số 47</b> Tính tích vô hướng của vectơ n chiều	120
<b>Bài số 48</b> Tính ma trận chuyển vị	123
<b>Bài số 49</b> Tìm các phần tử min và max trong một ma trận	126
<b>Bài số 50</b> Tính tổng hai ma trận	128
<b>Bài số 51</b> Nhân hai ma trận tổng quát	131

<b>Bài số 52</b> Nhân hai ma trận	135
<b>Bài số 53</b> Tính tích hai ma trận phức dùng pointer	138
<b>Bài số 54</b> Tính tích hai ma trận dùng file	142
<b>Bài số 55</b> Vào dữ liệu ma trận tổng quát dùng file	146
<b>Bài số 56</b> Tính giá trị định thức cấp n	149
<b>Bài số 57</b> Tìm ma trận nghịch đảo	153
<b>Bài số 58</b> Giải hệ phương trình đại số tuyến tính cấp n	158

## **Chương 5. Đa thức và nội suy đa thức**

<b>Bài số 59</b> Tính giá trị của một đa thức theo thuật toán Horner	164
<b>Bài số 60</b> Tính tổng của hai đa thức	166
<b>Bài số 61</b> Nhân hai đa thức	169
<b>Bài số 62</b> Chia hai đa thức	172
<b>Bài số 63</b> Tìm đa thức nội suy Lagrange	176
<b>Bài số 64</b> Tìm đa thức nội suy Newton	181
<b>Bài số 65</b> Tìm đa thức nội suy ma trận Lagrange	187
<b>Bài số 66</b> Tính đạo hàm cấp 1 của một đa thức	194
<b>Bài số 67</b> Tính đạo hàm cấp 1 của đa thức Newton	197

## **Chương 6. Tích phân số**

<b>Bài số 68</b> Tính tích phân xác định $\int_a^b f(x)dx$ bằng phương pháp Simpson	200
<b>Bài số 69</b> Tính tích phân xác định $\int_a^b f(x)dx$ bằng phương pháp hình thang	202

<b>Bài số 70</b> Tính tích phân kép $\iint f(x,y)dx dy$ bằng phương pháp Gauss-Legendre	204
<b>Bài số 71</b> Tính tích phân kép $\iint f(x,y)dx dy$ bằng phương pháp Romberg	208

## **Chương 7. Giải phương trình phi tuyến**

<b>Bài số 72</b> Tìm nghiệm trong khoảng $[a,b]$ của hàm phi tuyến bằng phương pháp chia đôi cung	213
<b>Bài số 73</b> Tìm nghiệm trong khoảng $[a,b]$ của hàm phi tuyến bằng phương pháp cát tuyến	215
<b>Bài số 74</b> Tìm nghiệm của hàm phi tuyến bằng phương pháp lập Newton	218

## **Chương 8. Tối ưu hóa**

<b>Bài số 75</b> Tìm cực trị của hàm một biến bằng phương pháp tiết diện vàng	222
<b>Bài số 76</b> Tìm cực trị của hàm một biến bằng phương pháp gradient	226
<b>Bài số 77</b> Tìm cực trị của hàm hai biến bằng phương pháp gradient	228

## **Chương 9 Phương trình vi phân**

<b>Bài số 78</b> Giải phương trình vi phân cấp một bằng phương pháp Euler	233
<b>Bài số 79</b> Giải phương trình vi phân cấp một bằng phương pháp Euler cải tiến	235

<b>Bài số 80</b> Giải phương trình vi phân cấp một bằng phương pháp Runge-Kutta	238
<b>Bài số 81</b> Giải hệ phương trình vi phân cấp n bằng phương pháp Runge-Kutta	241

## Chương 10. Xác suất và thống kê

<b>Bài số 82</b> Tính phân bố nhị thức	246
<b>Bài số 83</b> Tính phân bố Poisson	248
<b>Bài số 84</b> Tính kỳ vọng, phương sai, độ lệch chuẩn phương của các biến ngẫu nhiên	250
<b>Bài số 85</b> Cho hai bảng dữ liệu, tìm hệ số tương quan và mức tin cậy giữa các biến	252

## Chương 11. Một số chương trình xử lý văn bản, quản lý và một số bài toán ứng dụng

<b>Bài số 86</b> Lập chương trình in lịch thế kỷ	256
<b>Bài số 87</b> Lập chương trình kiểm tra các số liệu vào là số nguyên	260
<b>Bài số 88</b> Lập trình in tờ hóa đơn	263
<b>Bài số 89</b> Lập trình in các kiểu kẻ khung dùng ký tự đồ họa mã ASCII	266
<b>Bài số 90</b> Lập trình vào - ra một dữ liệu kiểu cấu trúc	274
<b>Bài số 91</b> Lập trình bài toán tháp Hà Nội dùng hàm đệ quy	277
<b>Bài số 92</b> Mảng một chiều kiểu cấu trúc	279
<b>Bài số 93</b> Mảng kiểu cấu trúc, danh sách từng người	282

<b>Bài số 94</b> Mảng kiểu cấu trúc, danh sách nhân sự	284
<b>Bài số 95</b> Mảng kiểu cấu trúc theo dòng, danh sách nhân sự	287
<b>Bài số 96</b> Cho ngày tháng sinh, tìm chòm sao tương ứng	291
<b>Bài số 97</b> Chương trình file truy nhập tuần tự	295
<b>Bài số 98</b> Chương trình file truy nhập tuần tự, cho phép tạo, sửa chữa, thêm vào cuối file và chọn tên file	300
<b>Bài số 99</b> Mã hóa và giải mã Morse một bức điện đánh từ bàn phím	309
<b>Bài số 100</b> Lập trình cho một bản nhạc. Áp dụng cho bài <i>Tiến quân ca</i> của nhạc sĩ Văn Cao	313
<b>Bài số 101</b> Tạo cửa sổ ở chế độ văn bản	319
<b>Phụ lục</b> Các hàm Turbo C thường gặp	321
<b>Tài liệu tham khảo</b>	327

## Chương 1

# MỘT SỐ CHƯƠNG TRÌNH MINH HỌA CƠ BẢN LẬP TRÌNH NGÔN NGỮ C

Chương này dành cho bạn đọc bắt đầu làm quen với ngôn ngữ C. Thông qua một số chương trình đơn giản nhằm minh họa cho những vấn đề cơ bản của việc lập trình ngôn ngữ C. 20 chương trình trong chương này có thuật toán đơn giản, chúng được xây dựng nhằm minh họa, giải thích việc sử dụng các toán tử chủ yếu của ngôn ngữ C.

**Bài số 1** Viết chương trình vào số liệu từ bàn phím và hiện kết quả trên màn hình cho các dữ liệu kiểu nguyên, thực và string.

**Bài giải** Chương trình CT1 gồm việc khai báo kiểu các dữ liệu kiểu nguyên, thực và một string gọi là ten[10], max là 10 ký tự.

Lưu ý là để vào dữ liệu từ bàn phím trong câu lệnh scanf ta phải dùng địa chỉ của biến &a, &b, &x mà không phải là giá trị của biến a, b, x. Phần điều khiển trong câu lệnh printf và scanf nằm giữa hai dấu nháy kép " và " phải có thứ tự đặc biệt phù hợp với kiểu của biến cần đưa vào hoặc hiện màn hình.

Nếu muốn in kết quả trên máy in thì câu lệnh in phải được viết như sau :

```
fprintf ( stdprn, " điều khiển ", danh sách biến cần in );
```

```

/* CT1 Chuong trinh minh hoa vao ra */
#include<stdio.h>
void main()
{
    int a; /* Khai bao kieu du lieu */
    long int b;
    float x;
    char ten[10]; /* Khai bao string max 10 ky tu */
    printf("\n Vao du lieu tu ban phim ");
    printf("\n a = "); scanf("%d",&a);
    printf("\n b = "); scanf("%ld",&b);
    printf("\n x = "); scanf("%f",&x);
    printf("\n Ten :");scanf("%s",ten);
    /* Hien thi ket qua */
    printf("\n a :%10d\n b :%10ld\n x :%10.2f",a,b,x) ;
    printf("\n Tac gia TURBOC :%s",ten);
    getch();
    return;
}

```

Ta đc kết quả chương trình CT1 như sau :

*Vao du lieu tu ban phim*

*a = - 12*

*b = 6*

*x = 16.82*

*Ten : BORLAND*

*a : - 12*

*b = 6*

*x = 16.82*

*Tac gia TURBO C : BORLAND*

### **Bài số 2 Tìm kết quả của chương trình sau đây ( CT2 )**

```

/* CT2 Chuong trinh minh hoa cach hien thi */
#include<stdio.h>
void main()
{
    int n = 543; /* Khai bao va khai dau cac bien */
    int p = 5;
    float x = 34.5678;
    printf("A:%d %f\n",n,x);
    printf("B:%4d %10f\n",n,x);
    printf("C:%2d %3f\n",n,x);
    printf("D:%10.3f %10d\n",x,n);
    printf("E:%-5d %f\n",n,x);
    printf("F:%*d\n",p,n);
    printf("G:%.*f\n",12,5,x);
    printf("H:%x :%8x :\n",n,n);
    printf("I:%o :%8o :\n",n,n);
    getch();
    return;
}

```

**Bài giải** Ta cần nắm vững ý nghĩa của các ký hiệu sau đây :

Ký hiệu	Ý nghĩa
d	dùng cho số nguyên hệ thập phân
o	dùng cho hệ cơ số 8

x	dùng cho hệ cơ số 16 ( từ 0 đến f )
X	dùng cho hệ cơ số 16 ( từ 0 đến F )
u	dùng cho số nguyên không dấu ( unsigned )
c	dùng cho char ( ký tự )
s	dùng cho string ( xâu ký tự )
e	số thực dạng khoa học m.nnnexx
E	số thực dạng khoa học m.nnnExx
f	số thực dạng mm.nn
z	tương tự như e hay f tùy theo giá trị cần chỉ thị

dấu âm ( - ) để căn sang bên trái, nếu không sẽ căn sang bên phải.

Kết quả của chương trình CT2 là :

A : 543 34.567799

B : 543 34.567799

C : 543 34.567799

D : 34.568 543

E : 543 34.567799

F : 543

G : 34.56780

H : 21f : 21f :

I : 1037 : 1037 :

**Bài số 3** Cho các biến kiểu thực và kiểu nguyên. Tìm giá trị gán cho các biến của chương trình CT3.

/\* CT3 Chương trình minh họa lệnh gan \*/

#include<stdio.h>

```
void main()
{
    int n = 5,p = 9; /* Khai bao va khai dau cac bien */
    int q1,q2,q3,q4,q5;
    float x1,x2,x3,x4;
    printf("\n Tim gia tri gan cho cac bien ");
    q1 = n<p;
    q2 = n == p;
    q3 = p %n + p>n;
    q4 = n*(p>n ? n :p);
    q5 = n*(p<n ? p :p);
    x1 = p/n;
    x2 = (float)p/n;
    x3 = (p +0.5)/n;
    x4 = (int)(p+0.5)/n;
    /* Hien thi gia tri cac bien */
    printf("\n q1 = %d ",q1);
    printf("\n q2 = %d ",q2);
    printf("\n q3 = %d ",q3);
    printf("\n q4 = %d ",q4);
    printf("\n q5 = %d ",q5);
    printf("\n x1 = %10.3f ",x1);
    printf("\n x2 = %10.3f ",x2);
    printf("\n x3 = %10.3f ",x3);
    printf("\n x4 = %10.3f ",x4);
    getch();
    return;
}
```

**Bài giải** Ta được kết quả chương trình CT3 là :

*Tìm giá trị gan cho các biến*

*q1 = 1*

*q2 = 0*

*q3 = 1*

*q4 = 25*

*q5 = 45*

*x1 = 1.000*

*x2 = 1.800*

*x3 = 1.900*

*x4 = 1.000*

**Bài số 4** Tìm kết quả của chương trình minh họa toán tử tự tăng tự giảm CT4.

```
/* CT4 Chuong trinh minh hoa toan tu tu tang,tu giam */
#include<stdio.h>
void main()
{
    int n = 10, p = 5,q = 10, r;
    printf("\n Chuong trinh minh hoa toan tu tu tang tu giam \n");
    r=n== (p=q);
    printf(" A :n =%d p =%d q =%d r =%d\n",n,p,q,r);
    n=p=q=5;
    n+=p+=q;
    printf(" B:n =%d p =%d q =%d \n",n,p,q);
    q=n<p?n++ :p++;
}
```

```

printf(" C :n =%d p =%d q =%d \n",n,p,q)
q=n>p?n++:p++;
printf(" D:n =%d p =%d q =%d \n",n,p,q);
getch();
return;
}

```

**Bài giải** Ta được kết quả chương trình CT4 là :

*Chuong trinh minh hoa toan tu tu tang tu giam*

*A : n = 10 p = 10 q = 10 r = 1*

*B : n = 15 p = 10 q = 5*

*C : n = 15 p = 11 q = 10*

*D : n = 16 p = 11 q = 15*

**Bài số 5** Tìm kết quả của chương trình toán tử logic.

```

/* CT5 Chuong trinh minh hoa toan tu logic */
#include<stdio.h>
void main()
{
int n,p,q;
printf("\n Chuong trinh minh hoa toan tu logic \n ");
n = 5; p = 2; /* Truong hop 1 */
q = (n++ >p )|| ( p++ != 3);
printf("A : n = %d p = %d q = %d\n",n,p,q);
n = 5; p = 2; /* Truong hop 2 */
q = (n++ <p )|| ( p++ != 3);
printf(" B : n = %d p = %d q = %d\n",n,p,q);

```

```

n = 5; p = 2; /* Truong hop 3 */
q = (++n == 3)&& (++p == 3);
printf(" C : n = %d p = %d q = %d\n",n,p,q);
n = 5; p = 2; /* Truong hop 4 */
q = (++n == 6)&& (++p == 3);
printf(" D : n = %d p = %d q = %d\n",n,p,q);
getch();
return;
}

```

**Bài giải** Toán tử **&&** và **||** chỉ tính toán hạng thứ hai khi cần. Ở đây trường hợp 1 và 3 không được tính. Kết quả chương trình CT5 là:

*Chuong trinh minh hoa toan tu logic*

A : n = 6 p = 2 q = 1  
B : n = 6 p = 3 q = 1  
C : n = 6 p = 2 q = 0  
D : n = 6 p = 3 q = 1

**Bài số 6** Cho mảng 3 phần tử kiểu nguyên và một con trỏ kiểu nguyên. Tìm kết quả của chương trình CT6.

```

/* CT6 Chuong trinh minh hoa bang va con tro */
#include<stdio.h>

void main()
{
    int i,j;
    int t[3]; /* Khai bao mang 3 phan tu */
    int * adt; /* Khai bao con tro kieu nguyen */

```

```

for (i = 0,j = 0; i < 3; i++) t[i] = j++ + i ; /* 1 */
printf("\n");
for (i = 0; i < 3 ;i++) printf(" %d", t[i]); /* 2 */
printf("\n");
for (i = 0; i < 3 ;i++) printf(" %d", *(t+i)); /* 3 */
printf("\n");
for (adt=t ;adt<t+3 ;adt++) printf(" %d",* adt); /* 4 */
printf("\n");
for (adt=t+2;adt>=t;adt--) printf(" %d",* adt); /* 5 */
getch();
return;
}

```

**Bài giải** Ta được kết quả của chương trình CT6 như sau :

0 2 4

0 2 4

0 2 4

4 2 0

**Bài số 7** Lập chương trình để phân loại học sinh theo điểm.

**Bài giải** Có nhiều phương án để rẽ nhánh chương trình này. Ở đây ta sử dụng câu lệnh if else để kiểm tra điểm của học sinh và rẽ nhánh chương trình.

Từ 0 đến 3 điểm thuộc loại kém.

4 điểm thuộc loại yếu.

Từ 5 đến 6 điểm thuộc loại trung bình.

Từ 7 đến 8 điểm thuộc loại khá.

Từ 9 đến 10 điểm thuộc loại giỏi.

Ngoài các điểm từ 0 đến 10 sẽ báo vào sai. Dùng lệnh nhảy với goto (nhấn tt) để tiếp tục chương trình.

```
/* CT7 Chương trình minh họa dùng if else */
#include<stdio.h>
void main()
{
    int diem;
    printf ("\n Chương trình phân loại học sinh theo điểm ");
    tt: printf ("\n VAO SO LIEU ");
    printf("\n Điểm = "); scanf("%d",&diem);
    if(( diem >= 0) && (diem <= 3))
        printf(" Kem \n");
    else if( diem == 4)
        printf(" Yeu \n");
    else if(( diem >= 5) && (diem <= 6))
        printf(" Trung bình \n");
    else if( diem >= 7 && diem <= 8)
        printf(" Kha \n");
    else if (( diem >= 9) && (diem <= 10))
        printf(" Gioi \n");
    else printf(" Vao sai \n");
    printf("\n Tiếp tục 1 / stop 0 : ");
    scanf(" %d", &diem);
    if ( diem == 1) goto tt;
    getch();
    return;
}
```

**Kết quả chương trình CT7 như sau :**

*Chương trình phân loại học sinh theo điểm*

*Vào số liệu*

*diem = 9*

*Gioi*

*tiếp tục 1 / 0 : 0*

**Bài số 8** Lập chương trình phân loại học sinh theo điểm sử dụng switch.

**Bài giải** Để phân nhánh chương trình ta sử dụng câu lệnh switch với các nhánh là kết quả điểm của học sinh.

```
/* CT8 Chương trình minh họa sử dụng switch */
#include<stdio.h>
void main()
{
    int diem;
    tt : printf(" Vào dữ liệu \n");
    printf("\n điểm = ") ; scanf("%d", &diem);
    switch(diem)
    {
        case 0:
        case 1:
        case 2:
        case 3: printf(" Kem \n ");break;
        case 4: printf(" Yếu \n ");break;
        case 5:
        case 6: printf(" Trung bình \n ");break;
    }
}
```

```

case 7;
case 8: printf(" Kha \n ");break;
case 9:
case 10: printf(" Gioi \n ");break;
default: printf(" Vao sai \n ");
}

printf("\n tiep tuc 1 / stop 0 :
scanf("%d", &diem);
if (diem == 1) goto tt;
getch();
return;
}

```

Kết quả chương trình CT8 là :

*Vao du lieu*

*diem = 4*

*Yeu*

*tiep tuc 1 / 0: 0*

**Bài số 9** Tìm tất cả các phương án để có được 1000đ từ các tờ giấy bạc 500đ, 200đ và 100đ.

**Bài giải** Ta dùng 3 chu trình for lồng vào nhau để thay đổi số tờ giấy bạc 500đ, 200đ và 100đ với điều kiện tổng của chúng bằng 1000đ.

```

/* CT9 Chuong trinh minh hoa su dung cau lenh for */
#include<stdio.h>
void main()
{
int sopan; /* bo dem so phuong an */

```

```

int t500, /* tien 500 d */
int t200; /* tien 200 d */
int t100; /* tien 100 d */

sopa = 0;
for ( t500 = 0; t500 <= 2; t500++)
    for ( t200 = 0; t200 <= 5; t200++)
        for ( t100 = 0; t100 <= 10; t100++)
            if ( 100*t100 + 200*t200 +500*t500 == 1000)
            {
                sopa++;
                printf("\n Tien 1000 d = ");
                if( t100 ) printf("    %2d X 100 d ",t100);
                if( t200 ) printf("    %2d X 200 d ",t200);
                if( t500 ) printf("    %2d X 500 d ",t500);
                printf("\n");
            }
printf("\n Co tat ca %d phuong an de co 1000 d ",sopa);
getch();
return;
}

```

**Kết quả ta được :**

*Tien 1000d = 10 X 100d*

*Tien 1000d = 8 X 100d 1 X 200d*

*Tien 1000d = 6 X 100d 2 X 200d*

*Tien 1000d = 4 X 100d 3 X 200d*

*Tien 1000d = 2 X 100d 4 X 200d*

*Tien 1000d = 5 X 200d*

*Tien 1000d = 5 X 100d 1 X 500d*  
*Tien 1000d = 3 X 100d 1 X 200d 1 X 500d*  
*Tien 1000d = 1 X 100d 2 X 200d 1 X 500d*  
*Tien 1000d = 2 X 500d*  
*Có tất cả 10 phương án để có 1000d*

**Bài số 10** Tìm kết quả của chương trình minh họa sử dụng While CT10.

```
/* CT10 Chuong trinh minh họa su dung while */
#include<stdio.h>
void main()
{
    int n,p;
    n = 0 ;
    printf("\n Chuong trinh minh họa su dung while \n ");
    while ( n <= 5 ) n++;
    printf("A : n = %d \n",n);

    n = p = 0;
    while ( n <= 8 ) n += p++;
    printf(" B : n = %d \n",n);
    n = p = 0;
    while ( n <= 8 ) n += ++p;
    printf(" C : n = %d \n",n);
    n = p = 0;
    while ( p <= 5 ) n += p++;
    printf(" D : n = %d \n",n);
    n = p = 0;
```

```

while ( p <= 5 ) n += ++p;
printf(" E : n = %d \n",n);
getch();
return;
}

```

**Bài giải** Kết quả của chương trình CT10 như sau :

**A : n = 6**

**B : n = 10**

**C : n = 10**

**D : n = 15**

**E : n = 21**

**Bài số 11** Tính tổng n số đầu tiên của cấp số

$$1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n}$$

**Bài giải** Dùng một chu trình for để tính tổng của n số đầu tiên. Trong đó biểu thức tong += ( float ) 1 / i ; được chuyển dạng thành float.

```

/* CT11 Chuong trinh minh hoa su dung do while */
void main()
{
    int nt; /* so thanh phan cua chuoi */
    float tong; /* tong cua chuoi */
    int i;
    do
    {
        printf("\ Cho so thanh phan n = ");
        scanf("%d",&nt);
    }

```

```

while (nt<1);
for ( i = 1,tong = 0;i <= nt; i++) tong += (float)1/i;
printf("\n Tong %d so dau tien = %f",nt,tong);
getch();
return;
}

```

Kết quả của chương trình CT11 là :

*Cho so thanh phan n = 5*

*Tong 5 so dau tien = 2.83334*

*Nhận xét :* Trong 3 thành phần cách nhau bằng dấu ; của câu lệnh for có thể vắng mặt nhiều thành phần hoặc có thể là một dãy lệnh. Ở CT11 trong thành phần thứ nhất của for là i = 1, tong =0;

### Bài số 12 Xây dựng hàm để in một xâu ký tự

**Bài giải** Ta xây dựng 3 hàm :

*void ham1 (void)* dùng in một string không có đối số và cũng không có giá trị trả về.

*void ham2 (int)* có đối số kiểu int và không có giá trị trả về và được gọi hai lần.

*int ham3 (int)* có đối số kiểu int và có giá trị trả về ( trả về 0 ) và được gọi 3 lần.

```

/* CT12 Chuong trinh minh hoa khai bao va goi ham */
#include<stdio.h>
void ham1 (void);
void ham2 (int);
int ham3 (int);

```

```

void main()
{
    printf("\n Chuong trinh minh hoa khai bao va goi ham \n");
    ham1();
    ham2(2); /* 2 lan goi ham */
    ham3(3); /* 3 lan goi ham */
    getch();
    return;
}

void ham1 (void)
{
    printf( " TURBOC \n");
}

void ham2 (int n)
{
    int i ;
    for ( i = 0; i < n; i++)
        printf( " Tac gia ngon ngu C \n");
}

int ham3 (int n)
{
    int i;
    for ( i = 0; i < n; i++)
        printf(" Dennis Ritchie \n");
    return 0;
}

```

Kết quả chương trình CT12 như sau :

*Chuong trinh minh hoa khai bao va goi ham*

*TURBO C*

*Tac gia ngon ngu C*

*Tac gia ngon ngu C*

*Dennis Ritchie*

*Dennis Ritchie*

*Dennis Ritchie*

**Bài số 13** Viết một hàm nhận hai đối số là số thực và một ký hiệu ứng với một trong bốn phép tính số học + - \* /

**Bài giải** Cần xây dựng một hàm gọi là SOHOC (float, float, char) để thực hiện các phép tính số học cộng, trừ, nhân, chia và ghi kết quả.

```
/* CT13 Chuong trinh minh hoa su dung ham */
#include<stdio.h>
float SOHOC (float ,float ,char); /* prototype */
void main()
{
    float x,y;
    printf("\n VAO DU LIEU ");
    printf("\n x = "); scanf("%f",&x);
    printf("\n y = "); scanf("%f",&y);
    printf("\n Tong la : %f ", SOHOC(x,y,'+'));
    printf("\n Hieu la : %f ", SOHOC(x,y,'-'));
    printf("\n Tich la : %f ", SOHOC(x,y,'*'));
    printf("\n Thuong la : %f ",SOHOC(x,y,'/'));
    getch();
    return;
}
```

```

float SOHOC ( float v1,float v2,char tinh ) /* dinh nghia ham */
{
    float ketqua;
    switch (tinh)
    {
        case '+' : ketqua = v1 + v2;
                     break;
        case '-' : ketqua = v1 - v2;
                     break;
        case '*' : ketqua = v1 * v2;
                     break;
        case '/' : ketqua = v1 / v2;
                     break ;
        default : ketqua = v1 + v2;
    }
    return ketqua;
}

```

Kết quả của chương trình CT13 là :

*VAO DU LIEU*

*x = 12*

*y = 24.5*

*Tong la : 36.500000*

*Hieu la : -12.500000*

*Tich la : 294.000000*

*Thuong la : 0.489796*

**Bài số 14** Cho mảng một chiều, vào dữ liệu mảng từ bàn phím và tính tổng của các phần tử của mảng.

**Bài giải** Dùng cho chu kỳ for để thay đổi chỉ số phần tử của mảng và vào số liệu phần tử của mảng. Dùng một chu kỳ for để tính tổng các phần tử của mảng.

```
* CT14 Chuong trinh minh hoa mang mot chieu */
#include<stdio.h>
void main()
{
    int i;
    float m[5],s;
    for ( i = 0; i < 4; ++i)
    {
        printf("\n m[%d] = ",i);scanf("%f", &m[i]);
    }
    s = 0;
    for ( i = 0; i <= 4;i++)
    {
        s = s+ m[i];
        printf("\n m[%d] = %8.2f",i,m[i]);
    }
    printf("\n Tong = %8.2f",s);
    getch();
    return;
}
```

Kết quả chương trình CT14 như sau :

*m[0] = -28*

*m[1] = 51.3*

*m[2] = 15*

*m[3] = 87.6  
m[4] = 21.2  
m[0] = -28.00  
m[1] = 51.30  
m[2] = 15.00  
m[3] = 87.60  
m[4] = 21.20  
Tong = 147.10*

Lưu ý phép toán lấy địa chỉ chỉ áp dụng được đối với các phần tử của mảng 1 chiều.

**Bài số 15** Cho mảng một chiều. Vào dữ liệu các phần tử của mảng và tính tổng của các phần tử của mảng, sử dụng con trỏ.

**Bài giải** Dùng con trỏ pm trỏ tới phần tử của mảng. Dùng chu trình for để vào các phần tử của mảng và tính tổng của các phần tử.

```
/* CT15 Chuong trinh minh hoa mang dung con tro */  
#include<stdio.h>  
void main()  
{  
    int i ;  
    float s, *pm, m[5];  
    pm = m ;  
    for ( i = 0; i < 5; i++ )  
    {  
        printf("\n m[%d] = ",i); scanf("%f", &pm[i]);  
    }  
    s = 0;  
    for ( i = 0; i < 5; i++ )
```

```

s += m[i];
printf("\n m[%d] = %8.2f",i,pm[i]);
}
printf("\n Tong la = %8.2f",s);
getch();
return;
}

```

Kết quả ta được :

*m[0] = 45*

*m[1] = 27*

*m[2] = 23*

*m[3] = 67.5*

*m[4] = 0*

*m[0] = 45.00*

*m[1] = 27.00*

*m[2] = 23.00*

*m[3] = 67.50*

*m[4] = 0.00*

*Tong = 162.50*

**Bài số 16** Viết chương trình đòi hỏi người sử dụng đưa vào một số nguyên nằm giữa 1 và 7 để hiện màn hình tên ngày trong tuần theo con số đưa vào ( 1: thứ hai, 2: thứ ba, . . . , 7: chủ nhật ).

**Bài giải** Ta tạo nên một mảng 7 con trỏ về một string ứng với tên mỗi ngày trong tuần. Vì xâu này là hằng nên có thể tạo nên một mảng được khởi đầu có dạng char \*ngay [7] = {"thứ hai", "thứ ba",...}; ngay[0] chứa địa chỉ của xâu có ký tự đầu tiên tức là "thứ hai", ngay [1] chứa

địa chỉ xâu ký tự "thu ba". Để hiện màn hình giá trị xâu ký tự i cần lấy địa chỉ ngay [i-1].

```
/* CT16 Chuong trinh minh hoa mang string */
#include<stdio.h>
void main()
{
    int i;
    char *ngay[7] = { " Thu hai "," Thu ba "," Thu tu ",
        " Thu nam"," Thu sau "," Thu bay "," Chu nhat "};
    printf("\n Chuong trinh minh hoa mang string ");
    do
    {
        printf("\n Cho mot so nguyen giua 1 va 7 :");scanf("%d",&i);
    }
    while ( i <= 0 || i > 7 );
    printf("\n Ngay so %d cua tuan la : %s",i,ngay[i-1]);
    getch();
    return;
}
```

Ta được kết quả chương trình CT16 như sau :

*Chuong trinh minh hoa mang string*

*Cho mot so nguyen giua 1 va 7 : 5*

*Ngay so 5 cua tuan la : Thu sau*

**Bài số 17** Lập chương trình đếm lần xuất hiện một ký tự trong một xâu ký tự.

**Bài giải** Chương trình cần khai báo một dòng nhiều nhất 128 ký tự và đọc xâu ký tự bằng hàm gets và tổ chức một chu trình while để đếm

số lần xuất hiện của một ký tự (trong CT17 là chữ a).

```
/* CT17 Chuong trinh minh hoa xu ly string */
#include<stdio.h>
#define HANG 128
void main()
{
    char dong[HANG]; /* khai bao mot dong 128 ky tu */
    int i; /* xet cac ky tu cua dong */
    int na; /* dem chu 'a' */
    printf("\n Cho mot dong chu : ");
    gets (dong);
    na= 0;
    i = 0;
    while(dong[i]) if (dong[i ++] == 'a') na++;
    printf("\n Dong co %d chu a ",na);
    getch();
    return;
}
```

Kết quả chương trình CT17 như sau :

*Cho mot dong chu : Cong hoa xa hoi chu nghia Viet Nam*

*Dong co 4 chu a*

**Bài số 18** Cho một mảng hai chiều, tìm phần tử âm đầu tiên trong mảng.

**Bài giải** Cần khai báo và khởi đầu mảng và dùng hai chương trình for để thay đổi chỉ số i,j của các phân tử. Dùng if ( $a[i][j] < 0$ ) để tìm phần tử âm.

```

/* CT18 Chuong trinh minh hoa mang nhieu chieu */
#include<stdio.h>
void main()
{
/* Khai bao va khai dau mang hai chieu */
int i,j
float a[3][4] = {
    { 4 , 2.5,6.2, -8 },
    { 3.2, 25, 7 , 0.5},
    { 1.5,-3 ,0, 5 }};

for ( i = 0; i < 3; i++)
    for ( j = 0; j < 4; j++)
        if ( a[i][j] < 0 ) goto kqua;
        printf("\ Mang khong co phan tu am ");
        goto end;
kqua : printf("\n Phan tu am dau tien la a[%d,%d] = %8.2f",
    i+1,j+1,a[i][j] );
end:
; /* Cau lenh rong */
getch();
return;
}

```

Kết quả chương trình CT18 như sau :

*Phan tu am dau tien la a[1,4] = -8.00*

**Bài số 19** Lập một cấu trúc để ghi ngày sinh, ngày tuyển dụng và  
bậc lương của cán bộ.

**Bài giải** Tổ chức một cấu trúc gọi là CANBO có trường là ngày sinh, ngày tuyển dụng, bậc lương và một cấu trúc gọi là DATE có trường là ngày tháng năm dùng cho ngày sinh và ngày tuyển dụng.

```
/* CT19 Chuong trinh minh hoa su dung cau truc */
#include<stdio.h>
void main()
{
    struct DATE /* Dinh nghia cau truc thoi gian */
    {
        /* Truong cua cau truc */
        int ng; /* Ngay */
        char t[10]; /* Thang la string nhieu nhat 10 ky tu */
        int nam; /* Nam */
    };
    struct CANBO /* Dinh nghia cau truc nhan su */
    {
        /* Truong cua cau truc */
        struct DATE dates; /* Ngay sinh */
        struct DATE datet; /* Ngay tuyen dung */
        float luong; /* Bac luong */
    };
    struct CANBO c;
    printf("\n Vao ngay sinh ngay thang nam : ");
    scanf(" %d%s%d", &c.dates.ng,c.dates.t,&c.dates.nam);
    printf("\n Vao ngay tuyen dung ngay thang nam : ");
    scanf(" %d%s%d",&c.datet.ng,c.datet.t,&c.datet.nam);
    printf("\n Tien luong : "); scanf("%f", &c.luong);
    printf("\n%4d%10s%6d",c.dates.ng,c.dates.t,c.dates.nam);
    printf("\n%4d%10s%6d",c.datet.ng,c.datet.t,c.datet.nam);
    printf("\n Luong = %.2f",c.luong);
```

```
getch();  
return;  
}
```

Kết quả chương trình CT19 như sau :

Vao ngay sinh ngay thang nam : 20 tam 1944

Vao ngay tuyen dung ngay thang nam : 15 muoi 1965

Tien luong : 650000

20 tam 1944

15 muoi 1965

Luong = 650000.00

**Bài số 20** Tìm chương trình thay thế các ký hiệu của chương trình CT20.

**Bài giải** Cần thận trọng trong việc dùng macro. Bộ tiền xử lý của chương trình biên dịch chỉ thực hiện xử lý văn bản mà không chú ý đến nội dung. Chương trình thay thế các ký hiệu của CT20 là :

```
void main ()  
{  
    int x,y,z;  
    x = 5;  
    y = 10*5;  
    printf("\n QUA LON");  
    z = x + y;  
    z = x + y + 6;  
    z = 5*x + y;  
    z = 5 * ( x + y );  
    z = 5 *( ( x ) + ( y ));  
    printf("\n z = %d",z);  
    getch();
```

```
return;
{
/* CT20 Chuong trinh minh hoa su dung macro */
#define BEGIN {
#define END }
#define INTEGER int
#define NB 10
#define LIMIT NB*5
#define SUMXY x+y
#define SUM1 (x+y)
#define SUM2 ((x)+(y))
#define MSG "\n QUA LON"
void main()
BEGIN
INTEGER x, y, z;
x=5;
y=LIMIT;
printf(MSG);
z=SUMXY;
z=SUMXY+6;
z=5*SUMXY;
z=5*SUM1;
z=5*SUM2;
printf("\n=%d",z);
getch();
return;
END
```

## Chương 2

# ĐẠI SỐ

Chương này trình bày một số chương trình thực hiện các phép tính đại số trên các số thực và số phức. Ở đây chỉ giới hạn trong lớp các bài toán đại số thường gặp trong các bài toán kỹ thuật và kinh tế.

**Bài số 21** Tính tổ hợp chập  $k$  trong  $n$  phần tử  $C_n^k$ .

**Bài giải** Tổ hợp chập  $k$  trong  $n$  phần tử ký hiệu  $C_n^k$  được định nghĩa như sau :

$$C_n^k = \frac{n!}{k!(n-k)!}$$

Ta có thể tính tổ hợp  $C_n^k$  trực tiếp từ định nghĩa bằng cách tính các gai thừa. Tuy nhiên ta cũng có thể tính tổ hợp đơn giản hơn bằng cách sử dụng công thức truy hồi :

$$C_n^k = \frac{n-k+1}{k} C_n^{k-1}, \quad \text{với } k = 1, 2, \dots, n$$

Ta tổ chức một chu trình nhằm tính lặp  $C_n^k$  theo  $C_n^{k-1}$ .

Chương trình gồm việc vào các số liệu sau : số các phần tử  $n$ , chập  $k$  và một chu trình for để tính tổ hợp.

```

/* CT21 Chuong trinh tinh to hop */
#include <stdio.h>
main()
{
int n,k,i;
float c;

printf("\n Chuong trinh tinh to hop chap k trong n phan tu");
printf("\n VAO SO LIEU \n");
printf(" n = ");scanf("%d",&n);
printf(" k = ");scanf("%d",&k);
c = 1;
for (i=1;i<=k;i++)
{
    c = (n-i+1)*c/i;
}
printf("To hop chap %d trong %d phan tu : %10.2f\n",k,n,c);
getch();
return;
}

```

Nếu vào các dữ liệu sau đây thì có kết quả chương trình :

*Chuong trinh tinh to hop chap k trong n phan tu*

*VAO SO LIEU*

*n = 10*

*k = 4*

*To hop chap 4 trong 10 phan tu : 210.00*

**Bài số 22** Tính tích của một dãy n số thực.

**Bài giải** Để tính tích của một dãy n số thực ta áp dụng công thức truy hồi :

$$P = P * x[i] , \text{ với giá trị đầu } P = 1$$

Chương trình bao gồm những bước sau đây :

- Đọc dãy n số thực, đây là một mảng n phần tử giá trị thực được đưa vào nhờ câu lệnh for.

- Tổ chức một chu trình tính tích và ghi kết quả.

```
/* CT22 Chuong trinh tinh tich n so thuc */
#include <stdio.h>
main()
{
    int i,n;
    float tich,x[100];
    printf(" Chuong trinh tinh tich cua n so thuc\n");
    printf(" VAO SO LIEU\n");
    printf("n = ");scanf("%d",&n);
    for (i=0;i<n;i++)
    {
        printf("x[%d] = ",i);
        scanf("%f",&x[i]);
    }
    tich=1;
    for (i=0;i<n;i++)
    {
        tich *=x[i];
    }
```

```
printf(" TICH SO = %10.3f\n ",tich);
getch();
return;
}
```

Trong chương trình này nếu đưa vào mảng dữ liệu như sau ta sẽ  
được kết quả :

*Chuong trinh tinh tích cua n so thuc*

*VAO SO LIEU*

*n = 4*

*x[0] = 12.52*

*x[1] = -45.6*

*x[2] = 25*

*x[3] = 50.45*

*TICH SO = -720062.750*

### **Bài số 23** Tính tổng và tích của dãy n số thực

**Bài giải** Thuật toán áp dụng cho bài này cũng tương tự bài 22

Chương trình gồm các bước sau :

- Nhập bảng dữ liệu n phần tử từ bàn phím
- Tổ chức một chu trình for để tính tổng và tính tích của n số thực theo các công thức lặp.

$$s = s + x[i]$$

$$p = p * p[i]$$

với bước ban đầu  $s = 0$  ,  $p = 1$ .

```

/* CT23 Chuong trinh tinh tong va tich */
#include <stdio.h>
main()
{
int i,n;
float p,s,e,x[10];

printf(" VAO SO LIEU \n");
printf("n = ");scanf("%d",&n);
for (i=1;i<=n;i++)
{
    printf("x[%d] = ",i);
    scanf("%f",&x[i]);
}
s = 0;
p = 1;
for (i=1;i<=n;i++)
{
    p *= x[i];
    s += x[i];
}
e = p+s;
printf("Ket qua Tich = %10.3f\n",p);
printf("Ket qua Tong = %10.3f\n",s);
printf("Ket qua Tong va Tich = %10.3f\n",e);
getch();
return;
}

```

Trong chương trình CT23 nếu ta vào các dữ liệu từ bàn phím sau đây sẽ có kết quả :

*VAO SO LIEU*

*n = 4*

*x[1] = 45.5*

*x[2] = 24.25*

*x[3] = 15*

*x[4] = 67.87*

*Ket qua Tich = 112391.000*

*Ket qua Tong = 152.602*

*Ket qua Tong va Tich = 1123443.625*

**Bài số 24** Tìm các số dương trong một dãy số

**Bài giải** Chương trình gồm việc vào dãy số từ bàn phím và dùng chu trình for để kiểm tra các giá trị  $x[i] > 0$  và ghi kết quả.

```
/* CT24 Chuong trinh tim so duong trong day so */
```

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
int i,n,t;
```

```
float x[100];
```

```
printf(" VAO SO LIEU \n");
```

```
printf("n = ");scanf("%d",&n);
```

```
for (i=0;i<n;i++)
```

```
{
```

```
printf("x[%d] = ",i);
```

```

scanf("%f",&x[i]);
}
t=0;
for (i=0;i<n;i++)
if (x[i]>0)
{
printf("x[%d] = %10.2f\n",i,x[i]);
t++;
}
printf("So duong = %d\n",t);
getch();
return;
}

```

Ta được kết quả chương trình CT24 như sau :

#### **VAO SO LIEU**

```

n = 5
x[0] = 23
x[1] = -15.68
x[2] = 38
x[3] = 490
x[4] = 57.65
x[0] =      23.00
x[2] =      38.00
x[3] =      490.00
x[4] =      57.65
So duong = 4

```

**Bài số 25** Tính trung bình số học của dãy n số thực trong một khoảng [a, b].

**Bài giải** Để tính trung bình số học của một dãy n số thực trong một khoảng [a, b] ta thực hiện một chu trình để kiểm tra các số x[i] có nằm trong khoảng [a, b], nếu đúng sẽ thực hiện tính tổng theo thuật toán đã trình bày trong bài số 23.

```
/* CT25 Chuong trinh tinh trung binh trong day so */
#include <stdio.h>
main()
{
    int i,n,t;
    float a,b,m,x[100];

    printf("\n Chuong trinh tinh trung binh\n");
    printf(" VAO SO LIEU\n");
    printf(" a = ");scanf("%f",&a);
    printf(" b = ");scanf("%f",&b);
    printf(" n = ");scanf("%d",&n);
    for (i=0;i<n;i++)
    {
        printf("x[%d] = ",i);scanf("%f",&x[i]);
    }
    m=0;
    t=0;
    for (i=0;i<n;i++)
    {
        if (x[i]>a)
            if (x[i]<b)
```

```

    {
        t++;
        m+=x[i];
    }
}

m=m/t;
printf(" TRUNG BINH TRONG KHOANG [a,b] = %f\n",m);
getch();
return;
}

```

Trong chương trình nếu cho các cận a,b và dãy n số thực như sau sẽ có kết quả :

*Chuong trinh tinh trung binh*

*a = 2*

*b = 10*

*n = 4*

*x[0] = 3.5*

*x[1] = 4.75*

*x[2] = -12.45*

*x[3] = 4.75*

*TRUNG BINH TRONG KHOANG [a,b] = 4.333333*

**Bài số 26** Tính 100 số đầu tiên của các số Fibonacci.

**Bài giải** Số Fibonacci được xác định theo công thức truy hồi sau :

$u_n = u_{n-1} + u_{n-2}$  ,  $n = 3,4 \dots ; u_1 = 1$  .  $u_2 = 2$

Ta thực hiện một chương trình for để tính lặp và ghi các số Fibonacci trong số n cho trước

```
/* CT26 Chuong trinh tinh chuoi FIBONACCI */
#include <stdio.h>
main()
{
int n,i,u[100];

printf("\n Chuong trinh tinh cac so FIBONACCI\n");
printf(" VAO SO LIEU \n");
printf(" n = ");scanf("%d",&n);
u[1]=1;
u[2]=2;
for (i=3;i<=n;i++)
{
    u[i]=u[i-1]+u[i-2];
    printf("%d ",u[i]);
}
getch();
return;
}
```

Kết quả với n = 10 ta có các số Fibonacci như sau :

*Chuong trinh tinh cac so FIBONACCI*

*VAO SO LIEU*

*n = 10*

*3 5 8 13 21 34 55 89*

**Bài số 27** Xác định số lớn nhất và số các số dương trong dãy n số thực.

**Bài giải** Gọi M = max x[i] và t là số các số dương trong dãy số đã cho. Đầu tiên coi M = x<sub>1</sub> và kiểm tra điều kiện x<sub>1</sub> > 0 để đếm các số dương, sau đó so sánh M với các giá trị x<sub>i</sub> để tìm giá trị lớn nhất. Thuật toán kết thúc khi đã kiểm tra hết n số. In giá trị lớn nhất M và số các số dương t.

```
/* CT27 Chuong trinh xac dinh max trong day so */
```

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
int t,i,n;
```

```
float max,x[100];
```

```
printf(" Chuong trinh xac dinh so lon nhat\n");
```

```
printf(" VAO SO LIEU\n");
```

```
printf(" n = ");scanf("%d",&n);
```

```
for (i=0;i<n;i++)
```

```
{
```

```
printf("x[%d] = ",i);
```

```
scanf("%f",&x[i]);
```

```
}
```

```
max = x[0];
```

```
t = 0;
```

```
for (i=0;i<n;i++)
```

```
{
```

```
if (x[i]>0) t = t + 1;
```

```
{
```

```

    if (max>x[i]) max = max;
    else max = x[i];
}
printf(" MAX = %f trong do co %d so la so duong\n",max,t);
getch();
return;
}

```

Trong chương trình nếu vào các dữ liệu sau đây sẽ có kết quả :

*Chuong trinh xac dinh so lon nhat*

*VAO SO LIEU*

*n = 4*

*x[0] = 10*

*x[1] = -25 57*

*x[2] = 120*

*x[3] = 23.65*

*MAX = 120.000000 trong do co 3 so la so duong*

**Bài số 28** Tính các phần tử của tam giác Pascal.

**Bài giải** Tam giác pascal được xác định như sau :

$$C_0^0$$

$$C_1^0 \ C_1^1$$

$$C_2^0 \ C_2^1 \ C_2^2$$

.....

$$C_n^0 \ C_n^1 \ C_n^2 \dots C_n^n$$

trong đó  $C_n^k$  là tổ hợp chập k trong n phần tử được xác định theo công thức truy hồi :

$$C_n^k = C_{n-1}^k + C_{n-1}^{k-1}$$

Cột đầu có tất cả các phần tử đều bằng 1, đồng thời giả thiết  $C_n^n = 1$ .

Trong chương trình có hàm `NhiThuc(int n, int p)` để thực hiện tính tổ hợp C theo công thức truy hồi.

```
/* CT28A Chuong trinh tinh tam giac PASCAL */
#include <stdio.h>
int NhiThuc(int,int); /* prototype */

main()
{
    int n,m,p;

    printf("\n Chuong trinh tinh tam giac Pascal\n");
    printf(" Cho so m = ");scanf("%d",&m);
    for (n=1;n<m+1;n++)
    {
        for (p=1;p<n+1;p++)
        {
            printf("%d ",NhiThuc(n,p));
        }
        printf("\n"); /* De xuong mot dong */
    }
    getch();
    return;
}
```

```

/* Ham Nhi thuc */
int NhiThuc(int n, int p)
{
    if (p==1 || p==n) return(1);
    else return(NhiThuc(n-1,p-1)+NhiThuc(n-1,p));
}

```

Kết quả nếu cho các dữ liệu vào như sau với chương trình CT28A:

*Chuong trinh tinh tam giac Pascal*

*Cho so m = 10*

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
1 9 36 84 126 126 84 36 9 1

```

*/\* CT28B Chuong trinh tinh tam giac PASCAL \*/*

```

#include <stdio.h>
#include <conio.h>
#define max 15

```

```

main()
{
int t[max], /* Bang mot dong cua tam giac */
n,          /* So dong thuc te */
i,j          /* Chi so dong, cot */
clrscr();
printf("Cho so dong n=");scanf("%d",&n);
if(n>max) n=max;
printf("\n\n p ");
for(i=0;i<n;i++)
{
    printf("%5d",i);
    printf("\n n\n");
for(i=0;i<=n;i++)
{
    printf("*****");
    printf("\n");
for(i=0;i<n;i++)
{
    {
        t[i]=1;
        for(j=i-1;j>0;j--)
            t[j]=t[j-1]+t[j];
        printf("%2d--",i);
        for(j=0;j<=i;j++)
            printf("%5d",t[j]);
        printf("\n");
    }
getch();
return;
}

```

Kết quả của chương trình CT28B :

Cho số dòng n = 10

p n	0	1	2	3	4	5	6	7	8	9
*****										
0 --	1									
1 --		1	1							
2 --		1	2	1						
3 --		1	3	3	1					
4 --		1	4	6	4	1				
5 --		1	5	10	10	5	1			
6 --		1	6	15	20	15	6	1		
7 --		1	7	21	35	35	21	7	1	
8 --		1	8	28	56	70	56	28	8	1
9 --		1	9	36	84	126	126	84	36	9

**Bài số 29** Đảo ngược một dãy n số thực.

**Bài giải** Thuật toán gồm việc đọc dãy n số thực a[i], i = 1...n và đảo ngược chỉ số của dãy i = n...1 và in kết quả của dãy.

```
/* CT29 Chuong trinh dao nguoc mot day so */
```

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
int i,j,n;
```

```
float tg,a[100];
```

```

printf("\n Chuong trinh dao nguoc mot day so\n");
printf(" Vao so con so cua day so n = ");
scanf("%d",&n);
for (i=1;i<=n;i++)
{
    printf(" a[%d] = ",i);
    scanf("%f",&a[i]);
}
printf("\n Day so dau tien\n");
for (i=1;i<=n;i++)
    printf("%5.2f ",a[i]);
for (i=1,j=n;i<j+i,i--) {
    tg = a[i];
    a[i] = a[j];
    a[j] = tg;
}
printf("\n Day so dao nguoc\n");
for (i=1;i<=n;+i)
    printf(" %5.2f ",a[i]);
getch();
return;
}

```

Trong chương trình nếu cho n dây số a; ta sẽ được kết quả như sau

*Chuong trinh dao nguoc mot day so*  
*Vao so con so cua day so n = 4*  
*a[1] = 12*

*a[2] = 35*

*a[3] = 46*

*a[4] = 78*

*Day so dau tien*

*12.00    35.00    46.00        78.00*

*Day so dao nguoc*

*78.00    46.00    35.00    12.00*

**Bài số 30** Sắp xếp dãy n số thực theo thứ tự tăng hoặc giảm

**Bài giải** Thuật toán sắp xếp một dãy số như sau :

- Đọc mảng n số thực.

- Để sắp xếp ta lấy phần tử đầu tiên và lần lượt so sánh với các phần tử tiếp theo nếu lớn hơn thì đổi chỗ cho giá trị của phần tử so sánh, tiếp tục kiểm tra và đổi chỗ cho đến khi đã kiểm tra hết n số thực và ghi kết quả dãy số đã sắp xếp.

```
/* CT30 Chuong trinh sap xep day so theo chieu giam */
```

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
int i,j,n;
```

```
float t,M[100];
```

```
printf("\n Chuong trinh sap xep mot mang n so\n");
```

```
printf("Cho n = ");scanf("%d",&n);
```

```
for (i=0;i<n;i++)
```

```
{
```

```
printf("M[%d] = ",i);
```

```
scanf("%f",&M[i]);
```

```
}
```

```

for (i=0;i<n;i++)
    for (j=0;j<n;j++)
    {
        if (M[i]>M[j])
        {
            t=M[i];
            M[i]=M[j];
            M[j]=t;
        }
    }
printf("\n Ket qua sau khi sap xep\n");
for (i=0;i<n;i++)
    printf("\nM[%d] = %f ",i,M[i]);
getch();
return;
}

```

Trong chương trình nếu lấy n = 5 và cho dãy số m<sub>i</sub> như sau sẽ có kết quả :

*Chương trình sap xep mot mang n so*

*Cho n = 4*

*M[0] = 12*

*M[1] = 45*

*M[2] = 67*

*M[3] = 89*

*Kết quả sau khi sap xep*

*M[0] = 89.000000*

$M[1] = 67.000000$

$M[2] = 45.000000$

$M[3] = 12.000000$

**Bài số 31** Giải phương trình trùng phương  $ax^4 + bx^2 + c = 0$

**Bài giải** Phương trình trùng phương được giải theo thuật toán sau:

$$\text{delta} = b^2 - 4ac$$

Nếu  $\text{delta} > 0$  có 2 nghiệm:  $t_1 = (-b + \sqrt{\text{delta}}) / (2*a)$

$$t_2 = (-b - \sqrt{\text{delta}}) / (2*a)$$

nếu  $t_1 \geq 0$  có nghiệm thứ nhất  $x_1 = \sqrt{t_1}$

có nghiệm thứ hai  $x_2 = -\sqrt{t_1}$

nếu  $t_2 \geq 0$  nghiệm thứ ba  $x_3 = \sqrt{t_2}$

nghiệm thứ tư  $x_4 = -\sqrt{t_2}$

nếu không sẽ vô nghiệm.

Nếu  $\text{delta} = 0$  có nghiệm kép  $t = -b/2a$

nghiệm thứ nhất  $x_1 = \sqrt{t_1}$

nghiệm thứ hai  $x_2 = -\sqrt{t_1}$

Nếu  $\text{delta} < 0$  phương trình vô nghiệm.

```
/* CT31 Chuong trinh giai phuong trinh trung phuong */
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
void main ()
```

```
{
```

```
float a,b,c,delta,t1,t2,tg;
```

```
printf("\n Giai phuong trinh trung phuong\n");
```

```
printf("\n Vao he so a = "); scanf("%f",&a);
```

```

printf("\n Vao he so b = "); scanf("%f",&b);
printf("\n Vao he so c = "); scanf("%f",&c);
delta=b*b-4*a*c;
if (delta>=0)
{
if (delta>0) /* Delta duong */
{
t1=(-b+sqrt(delta))/(2*a);
t2=(-b-sqrt(delta))/(2*a);
if (t1>=0)
{
tg=sqrt(t1);
printf(" Nghiem thu nhat x1 = %8.3f\n",tg);
printf(" Nghiem thu hai x2 = %8.3f\n",-tg);
}
if (t2>=0)
{
tg=sqrt(t2);
printf(" Nghiem thu ba x3 = %8.3f\n",tg);
printf(" Nghiem thu tu x4 = %8.3f\n",-tg);
}
if ((t1<0)&&(t2<0))
printf("Phuong trinh vo nghiem\n ");
} /* delta>0 */
else if (delta==0)
{
t1 = -b/(2*a);
if (t1>=0)

```

```

    {
        tg=sqrt(t1);
        printf (" Nghiem thu nhat x1 = %8.3f\n", tg);
        printf (" Nghiem thu nhat x2 = %8.3f\n", -tg);
    }
else
    printf(" Phuong trinh vo nghiem\n");
} /* delta = 0*/
}
else if(delta<0)
printf("Phuong trinh vo nghiem\n");
getch ();
return;
}

```

Trong chương trình 31 nếu cho các dữ liệu như sau :

*Giai phuong trinh trung phuong*

Vao he so a = 2.5

Vao he so b = -8.65

Vao he so c = 1.25

*Nghiem thu nhat x1 = 1.819*

*Nghiem thu hai x2 = -1.891*

*Nghiem thu ba x3 = 0.389*

*Nghiem thu tu x4 = - 0.389*

**Bài số 32** Gọi a, b, c là ba cạnh của một tam giác là các số dương bất kỳ. Nếu ba số đó có thể là các cạnh của tam giác hãy tính diện tích của

tam giác đó. Thủ điều kiện đó là tam giác thường hay tam giác vuông, tam giác cân hoặc tam giác đều.

### Bài giải Thuật toán như sau :

- Đọc ba số dương a,b,c bất kỳ, nếu chúng là các cạnh của một tam giác thì chiều dài của một cạnh phải nhỏ hơn tổng chiều dài của hai cạnh kia. Nếu đúng sẽ tính nửa chu vi của tam giác theo công thức :

$$p = \frac{a + b + c}{2}$$

và tính diện tích tam giác theo công thức Heron :

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

- Tam giác là vuông nếu thỏa mãn định lý Pitago  $c^2 = a^2 + b^2$
- Tam giác là đều khi  $a = b = c$ , là cân khi  $a = b$ , hoặc  $b = c$ , hoặc  $c = a$
- Trong chương trình nếu không thỏa mãn điều kiện các cạnh của một tam giác sẽ ghi dòng chữ KHONG TAO THANH TAM GIAC

```
/* CT32 Chương trình tính tam giác */
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
main()
```

```
{
```

```
float a,b,c,p,s;
```

```
printf("\n\n\tChương trình tính tam giác\n");
```

```
printf("Vui lòng nhập các cạnh của tam giác:\n");
```

```
printf("a = ");scanf("%f",&a);
```

```
printf("b = ");scanf("%f",&b);
```

```
printf("c = ");scanf("%f",&c);
```

```

if((a>0)&&(b>0)&&(c>0)&&(a<(b+c))&&(b<(a+c))&&(c<(a+b)))
{
    printf("\n Ba so da cho la ba canh cua tam giac\n");
    if ((a==b) || (b==c) || (a==c))
        if ((a==b)&&(b==c))
            printf("\n Day la tam giac deu\n");
        else printf("\n Day la tam giac can\n");
        if ((a*a+b*b==c*c) || (b*b+c*c==a*a) || (a*a+c*c==b*b))
            printf("\n Day la tam giac vuong\n");
    p=(a+b+c)/2;
    s=sqrt(p*(p-a)*(p-b)*(p-c));
    printf(" Chu vi tam giac la : %8.2f\n",2*p);
    printf(" Dien tich tam giac la : %8.2f\n",s);
}
else
printf(" KHONG TAO THANH TAM GIAC\n");
getch();
return;
}

```

Nếu cho các cạnh như sau sẽ có kết quả :

*Chuong trinh tam giac*

*Vao a,b,c la cac canh cua tam giac*

*a = 4*

*b = 3*

*c = 5*

*Ba so da cho la ba canh cua tam giac*

*Day la tam giac vuong*

*Chu vi tam giac la : 12.00*

*Dien tich tam giac la : 6.00*

### **Bài số 33** Tính giai thừa n! của một số nguyên

**Bài giải** Theo công thức định nghĩa của giai thừa

$$n! = n(n - 1)(n - 2)\dots 1$$

ta sẽ lập một chu trình tính tích theo công thức truy hồi

$p = p \cdot k$  với  $k$  là số nguyên, khi  $k > n$  sẽ ra khỏi chu trình và ghi kết quả  $p$ .

Ta cũng có thể tính giai thừa theo công thức đệ quy bằng cách định nghĩa hàm GTDe quy (n) và hàm Giai Thua (n) theo công thức truy hồi.

Chương trình tổ chức tính giai thừa theo công thức lặp và theo giai thừa đệ quy.

```
/* CT33 Chuong trinh tinh giai thua */
#include <stdio.h>
long GiaiThua(int);
long GTDequy(int);

main()
{
    int n;
    tt: printf("\n Cho mot so nho hon 16 n = ");scanf("%d",&n);
    printf("\n Giai thua cua %d la : %20ld\n",n,GiaiThua(n));
    printf("\n Giai thua de quy cua %d :%20ld\n",n,GTDequy(n));
    printf(" Tiep tuc hoac stop (go 1 hoac 0) :");scanf("%d",&n);
    printf("\n");
```

```

if (n==1) goto tt;
getch();
return;
}
long GiaiThua(n)
int n;
{
long s=1,i;
if (n==1 || n==0)
    return(1);
else
{
    for (i=1;i<=n;++i) s *=i;
    return(s);
}
}

long GTDequy(n)
int n;
{
if (n==1||n==0) return(1);
else return(n*GTDequy(n-1));
}

```

Nếu cho n ta được kết quả như sau :

*Cho mot so nho hon 16 n = 10*

*Giai thua cua 10 la : 3628800*

*Giai thua de quy cua 10 : 3628800*

*Tiep tuc hoac stop ( go 1 hoac 0 ) : 0*

**Bài số 34** Xác định các số nguyên tố trong khoảng từ 1 đến n với n là số tự nhiên cho trước.

**Bài giải** Có nhiều thuật toán khác nhau để tìm các số nguyên tố trong khoảng 1 đến n.

Thuật toán của Eratosthène như sau :

Từ tập các số nguyên từ 1 đến n mỗi lần gấp một số nguyên tố sẽ bị loại trừ ra khỏi tập này tất cả các số là bội của số nguyên tố này. Eratosthène là nhà toán học từ thời chưa có giấy viết. Vì vậy bài toán tìm số nguyên tố được ông đặt ra và thực hiện trên mặt đất bằng cách kẻ một dãy ô trong đó ghi các số tự nhiên từ 1 đến n. Số nào không phải là số nguyên tố thì bị loại ra bằng cách chọc một lỗ vào ô đó để xóa đi. Kết quả trên mặt đất lỗ chổ như một cái sàng lọc các số, vì thế bài toán có tên là *Sàng Eratosthène*.

Chương trình CT34A xác định các số nguyên tố từ 1 đến n.

```
/* CT34A Sang Eratosthene dynamic */
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#define true 1
#define false 0

main()
{
    unsigned n, /* So luong cac so nguyen */
        *Gachdi, /* con tro cac so khong ngto */
        soNgto,
        i;
    int na;      /* Bo dem so cua sang */
```

```

/* Cho n va dinh vi mang */
clrscr();
do
{
    printf("\nCho so luong cac so can xet n = ");
    scanf("%u",&n);
    Gachdi = (unsigned *)malloc((n+1)*sizeof(unsigned));
    if (Gachdi == NULL)
        printf("\nKhong du bo nho\n");
}
while (Gachdi == NULL);
/* Khoi tao sang */
for (i=1;i<=n;i++) /* Gan cac so 0 cho sang */
    Gachdi[i] = false;
Gachdi[1] = true; /* Gach so 1 vi khong la so nguyen to */
/* Chuyen vao sang */
soNgto = 1;
while (soNgto*soNgto<=n)
{
    while (Gachdi[++soNgto] && soNgto<n) {}
    /* Tim so dau tien khong bi Gach di */
    for (i=2*soNgto;i<=n;i+=soNgto) /* Gach cac boi so cua so
ngto */
        Gachdi[i] = true;
}
/* Hien thi ket qua */
printf("\nGiua 1 va %d co cac so nguyen to la : \n\n",n);
na = 0;

```

```

for (i=1;i<=n;i++)
if ( !Gachdi[i] /* Khong nam trong mang gach di - do la so
ngto */
{
    printf("%6u",i);
    na++ ;
    if(na%150==0) /* Mot trang man hinh chua 15 dong = 150 so
ng to */
    {
        printf("\n"); /* Xuong dong sau khi het mot trang */
        printf("An mot phim bat ky de xem tiep\n");
        getch();
    }
    else
    if (na%10==0) printf("\n"); /* 10 so mot dong */
}
printf("\nTrong khoang tu 1 den %d co %d so nguyen to.",n,na);
getch();
return;
}

```

Nếu cho n ta được kết quả như sau :

*Cho so luong cac so can xet n = 25*

*Giua 1 va 25 co cac so nguyen to la :*

*2 3 5 7 11 13 17 19 23*

*Trong khoang tu 1 den 25 co 9 so nguyen to.*

Chương trình CT34B xác định các số nguyên tố trong một khoảng [n1,n2]

```
/* CT34B Chuong trinh Sang Eratosthene */
#include <stdio.h>
#include<conio.h>
main()
{
int i,j,n1,n2,na;
clrscr();
printf("\nChuong trinh tim so nguyen to giua khoang 2 so\n");
printf("\nVao so tu nhien thu nhat n1 = ");
scanf("%d",&n1);
printf("\nVao so tu nhien thu hai n2 (n2<=32767) = ");
scanf("%d",&n2);
na = 0;
for (i=n1;i<=n2;i++)
{
    for (j=2;j<=i-1;j++)
        if (i%j==0) goto aaa;
    printf("%d\t",i);
    na++;
    if (na%150==0)      /* Mot trang man hinh chua 150 so
nguyen to */
    {
        printf("\n"); /* Xuong dong sau khi het mot trang */
        printf("An mot phim bat ky de xem tiep\n");
        getch();
    }
}
```

```

aaa;
}
getch();
return;
}

```

Kết quả chương trình sẽ là :

*Chuong trinh tim so nguyen to giua khoang 2 so  
Vao so tu nhien thu nhat n1 = 10  
Vao so tu nhien thu hai n2 ( n2 < = 32767 ) = 25  
11 13 17 19 23*

**Bài số 35** Lập chương trình thực hiện các phép tính cộng, trừ, nhân, chia, khai căn bậc hai và lũy thừa bậc n của số phức

**Bài giải** Cũng như ngôn ngữ PASCAL, ở ngôn ngữ C không có định nghĩa số phức. Tuy vậy chúng ta cũng có thể thực hiện các phép tính đối với các số phức bằng cách dùng các cấu trúc có dạng :

```

struct {
    float re;
    float im;
} sophuc;

```

Ta xây dựng các hàm thực hiện các phép tính đối với số phức theo thuật toán sau :

$$\text{Cộng trừ } (x_1 \pm iy_1) + (x_2 \pm iy_2) = (x_1 \pm x_2) + i(y_1 \pm y_2)$$

$$\text{Nhân } (x_1 \pm iy_1)(x_2 \pm iy_2) = (x_1x_2 - y_1y_2) + i(x_1y_2 + x_2y_1)$$

$$\text{Chia } \frac{x_1 + iy_1}{x_2 + iy_2} = \frac{x_1x_2 + y_1y_2}{x_2^2 + y_2^2} + \frac{i(y_1x_2 - x_1y_2)}{x_2^2 + y_2^2}$$

Căn bậc hai  $x + iy = (a + ib)^{1/2}$ , trong đó  
 $x = [a/2 + ((a/2)^2 + (b/2)^2)^{1/2}]^{1/2}$   
 $y = [-a/2 + ((a/2)^2 + (b/2)^2)^{1/2}]^{1/2}$

nếu  $a > 0$  ta tính  $x$  và  $y = b/2x$

nếu  $a < 0$  ta tính  $y$  và  $x = b/2y$

Luỹ thừa  $(a + ib)^n = r^n \cdot e^{in\theta} = r^n (\cos n\theta + i \sin n\theta)$

Trong đó  $r = (a^2 + b^2)^{1/2}$

$$\theta = \operatorname{arctg} \frac{b}{a}$$

Trong chương trình CT35 gồm tổ chức thực hiện 5 hàm sau :

- Tổng hai số phức
- Tích hai số phức
- Thương hai số phức
- Căn bậc hai của một số phức
- Luỹ thừa bậc n của một số phức

Chương trình chính gồm việc đưa các số phức cần tính, gọi 5 hàm và ghi kết quả.

```
/* CT35 Chương trình tính toán số phức */
#include <stdio.h>
#include <math.h>
typedef struct {      /* cấu trúc số phức */
    float re;
    float im;
} sophuc;

void TONG(sophuc,sophuc,sophuc); /* prototype hàm phuc */
void HIEU(sophuc,sophuc,sophuc);
void TICH(sophuc,sophuc,sophuc);
```

```
void THUONG(sophuc,sophuc,sophuc);
void CAN(sophuc,sophuc);
void LuyThua(sophuc,sophuc);

main ()
{
sophuc x,y,z;
printf(" Vao so phuc x\n");
printf("x.re = ");
scanf("%f",&x.re);
printf("x.im = ");
scanf("%f",&x.im);
printf(" Vao so phuc y\n");
printf("y.re = ");
scanf("%f",&y.re);
printf("y.im = ");
scanf("%f",&y.im);
z.re=0;
z.im=0;
TONG(x,y,z);
TICH(x,y,z);
THUONG(x,y,z);
CAN(x,z);
LuyThua(x,z);
getch();
return;
}
```

```

void TONG(sophuc a,sophuc b,sophuc c)
{
c.re=a.re+b.re;
c.im=a.im+b.im ;
printf(" Tong la (%.6lf) + j%.5lf\n",c.re,c.im);
return;
}

void HIEU(sophuc a,sophuc b,sophuc c)
{
c.re=a.re-b.re;
c.im=a.im-b.im;
printf(" Hieu la (%.6lf) + j%.5lf\n",c.re,c.im);
return;
}

void TICH(sophuc a,sophuc b,sophuc c)
{
c.re=a.re*b.re-a.im*b.im;
c.im=a.re*b.im+a.im*b.re;
printf(" Tich la (%.6lf) + j%.5lf\n",c.re,c.im);
return;
}

void THUONG(sophuc a,sophuc b,sophuc c)
{
c.re=(a.re*b.re+a.im*b.im)/(b.re*b.re+b.im*b.im);
c.im=(a.im*b.re-a.re*b.im)/(b.re*b.re+b.im*b.im);
printf(" Thuong la (%.6lf) + j%.5lf\n",c.re,c.im);
}

```

```

return;
}

void CAN(sophuc a,sophuc c)
{
printf("Khai can mot so phuc : \n");
printf("Cho so phuc : \n");
printf("co phan thuc : ");
scanf("%f",&a.re);
printf("co phan ao : ");
scanf("%f",&a.im);
if (a.re>0)
{
    c.re=sqrt(a.re/2+sqrt(((a.re/2)*(a.re/2))+((a.im/2)*(a.im/2))));
    c.im=a.im/(2*c.re);
}
else
{
    c.im=sqrt(-a.re/2+sqrt(((a.re/2)*(a.re/2))+((a.im/2)*(a.im/2))));
    c.re=a.im/(2*c.im);
}
printf(" Can la (%.6.1f) + j%.5.1f\n",c.re,c.im);
return;
}

void LuyThua(sophuc a,sophuc c)
{
int n;
float r,t;

```

```

printf("Luy thua mot so phuc : \n");
printf("Cho so phuc : \n");
printf("co phan thuc : ");
scanf("%f",&a.re);
printf("co phan ao : ");
scanf("%f",&a.im);
printf("Cho bac cua luy thua n = ");
scanf("%d",&n);
r = sqrt((a.re*a.re)+(a.im*a.im));
t = atan(a.im/a.re);
c.re = pow(r,n)*cos(n*t);
c.im = pow(r,n)*sin(n*t);
printf("Luy thua bac %d cua so phuc (%f)+j(%f) la\n",n,a.re,a.im);
printf(" (%f + j(%f)\n",c.re,c.im);
return;
}

```

*Vao so phuc x*

*x.re = 5*

*x.im = -7.5*

*Vao so phuc y*

*y.re = 6.75*

*y.im = 1.25*

*Tong la ( 11.8 ) + j ( -6.2 )*

*Tich la ( 43.1 ) + j ( -44.4 )*

*Thuong la ( 0.5 ) + j ( -1.2 )*

*Khai can mot so phuc ;*

*Cho so phuc :*

*co phan thuc : 5.75*

*co phan ao : -12.25*

*Cau la ( 3.1 ) + j ( -2.0 )*

*Luy thua mot so phuc :*

*Cho so phuc :*

*co phan thuc : 8.25*

*co phan ao : -7.15*

*Cho bac cua luy thua n = 3*

*Luy thua bac 3 cua so phuc ( 8.250000 ) + j( -7.150000 ) la*

*( -703.766357 ) + j ( -1094.414795 )*

**Bài số 36** Giải phương trình bậc hai với a,b,c là các số phức dạng

$$ax^2 + bx + c = 0$$

**Bài giải** Ta áp dụng thuật toán giải phương trình bậc hai quen thuộc với điều kiện các hệ số a,b,c và nghiệm được biểu diễn dưới dạng cấu trúc có hai thành phần thực và ảo như trong bài số 35

```
/* CT36 Giai phuong trinh bac hai he so phuc */
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
typedef struct {
```

```
    float real;
```

```
    float imag;
```

```
} sophuc;
```

```
main ()
```

```
{
```

```
sophuc a,b,c,s1,s2,u,v,w;
```

```
double r;
```

```

printf("Vao so phuc a\n");
printf("a.real = ");
scanf("%f",&a.real);
printf("a.imag = ");
scanf("%f",&a.imag);
printf("Vao so phuc b\n");
printf("b.real = ");
scanf("%f",&b.real);
printf("b.imag = ");
scanf("%f",&b.imag);
printf("Vao so phuc c\n");
printf("c.real = ");
scanf("%f",&c.real);
printf("c.imag = ");
scanf("%f",&c.imag);
u.real=b.real*b.real-b.imag*b.imag-4*a.real*c.real+4*a.imag*c.imag;
u.imag = 2*b.real*b.imag-4*a.real*c.imag-4*a.imag*c.real;
r = sqrt(u.real*u.real+u.imag*u.imag);
v.real = sqrt((r+u.real)/2);
v.imag = (u.imag<0)?-sqrt((r-u.real)/2) :sqrt((r-u.real)/2);
w.real = (-b.real - v.real) / 2;
w.imag = (-b.imag - v.imag) / 2;
u.real = (-b.real + v.real) / 2;
u.imag = (-b.imag + v.imag) / 2;
r = a.real*a.real + a.imag*a.imag;
s1.real = (a.real*w.real + a.imag*w.imag) / r;
s1.imag = (a.real*w.imag - a.imag*w.real) / r;
s2.real = (a.real*u.real + a.imag*u.imag) / r;

```

```

s2.imag = (a.real*u.imag - a.imag*u.real) / r;
printf("Phuong trinh co hai nghiem la\n");
printf("s1 = (%.6.2f) + j%.6.2f\n",s1.real,s1.imag);
printf("s2 = (%.6.2f) + j%.6.2f\n",s2.real,s2.imag);
getch();
return;
}

```

Kết quả nếu ta đưa các dữ liệu vào CT36 sẽ có kết quả như sau :

*Vao so phuc a*

*a.real = 1.75*

*a.imag = 3.25*

*Vao so phuc b*

*b.real = -4.12*

*b.imag = 5.75*

*Vao so phuc c*

*c.real = 1.25*

*c.imag = 2.15*

*Phuong trinh co hai nghiem la*

*s1 = (-0.10) + j( 0.30 )*

*s2 = (-0.74) + j(-2.02 )*

**Bài số 37** Tính tổng và tích của hai số phức dùng con trỏ

**Bài giải** Cũng như các bài toán đối với số phức, ta cần định nghĩa một cấu trúc kiểu

```

struct
{
    double real;
    double imag;
} sophuc;
```

Để thuận tiện ta xây dựng hai hàm tính tổng và tích có biến là các con trỏ dạng :

```

void TinhTong ( sophuc * , sophuc * , sophuc * )
void TinhTich ( sophuc * , sophuc * , sophuc * )

/* CT37 Chuong trinh tong va tich hai phuc so dung pointer */
#include<stdio.h>
typedef struct
{
    double real;
    double imag;
} sophuc;

void TinhTong(sophuc *,sophuc *,sophuc *);
void TinhTich(sophuc *,sophuc *,sophuc *);

main()
{
    sophuc z1,z2,tong,tich;
    z1.real = 0.5; z1.imag = 1.0;
    z2.real = 1.0; z2.imag = 1.0;
    TinhTong(&z1,&z2,&tong);
    TinhTich(&z1,&z2,&tich);
    printf("Hai so phuc (%.2f) + j%.2f) va (%.2f) + j%.2f)\n",
        z1.real,z1.imag,z2.real,z2.imag);
    printf("co tong la (%.2f) + j%.2f) \n",tong.real,tong.imag);
    printf("co tich la (%.2f) + j%.2f) \n\n",tich.real,tich.imag);
```

```

printf(" an Enter de tiep tuc \n");
getch(); /* Dung man hinh de xem ket qua */
return; /* Phai co cau lenh nay neu dung TC version 2.0 */
}

void TinhTong(sophuc *x,sophuc *y,sophuc *tong)
{
    tong->real = x->real + y->real;
    tong->imag = x->imag + y->imag;
}

void TinhTich(sophuc *x,sophuc *y,sophuc *tich)
{
    tich->real = x->real * y->real - x->imag * y->imag;
    tich->imag = x->real * y->imag + x->imag * y->real;
}

```

Các dữ liệu đưa vào trong chương trình và cho ta kết quả như sau:

*Hai so phuc (0.50) + j(1.00) va (1.00) + j(1.00)*

*co tong la (1.50) + j(2.00)*

*co tich la (-0.50) + j(1.50)*

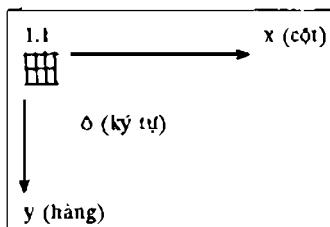
*an Enter de tiep tuc*

## Chương 3

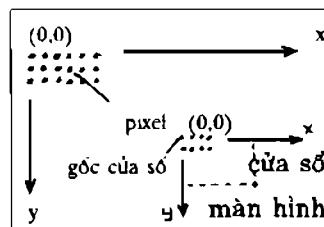
# ĐỒ HỌA

Khả năng đồ họa phong phú là một trong những ưu điểm nổi bật của ngôn ngữ C. Chương này sẽ giới thiệu một số chương trình về các hàm giải tích trong mặt phẳng và trong không gian.

Trước hết ta cần phân biệt màn hình ở chế độ đồ họa. Ở chế độ văn bản màn hình được tạo bởi các ô thường có 25 hàng và 80 cột. Mỗi ô biểu diễn cho một ký tự. Góc trái phía trên ứng với ô (1,1) và số cột theo chiều từ trái sang phải, số hàng theo chiều từ trên xuống dưới (hình 3-1).



Chế độ văn bản



Chế độ đồ họa

Hình 3-1

Ở chế độ đồ họa, màn hình gồm ma trận các điểm sáng (pixel). Mỗi điểm có khả năng tạo nên một hay nhiều màu. Tùy theo chế độ đồ họa và loại màn hình sử dụng chiều rộng và chiều cao của ma trận điểm sáng có thể là 640x350 (EGA 16 Color mode), 640x480 (VGA 16 Color mode). Gốc tọa độ ở góc trên phia trái, chiều ox từ trái sang phải, chiều oy từ trên xuống dưới. Cửa sổ màn hình được xác định bằng setviewport là hình chữ nhật chứa mọi đường vẽ. Các đường nằm ngoài cửa sổ này sẽ bị xén mất. Trên hình 3-1 vẽ cửa sổ màn hình ở giữa màn

Để xây dựng một chương trình đồ họa đầu tiên ta phải khởi tạo hệ thống đồ họa, xác định loại màn hình, một đồ họa sẽ sử dụng. Tiếp theo cần xác định màu nền, màu đường vẽ, màu tô. Cuối cùng là biểu diễn các hàm giải tích theo các tọa độ sao cho có thể sử dụng được các hàm của chế độ đồ họa. Khởi tạo chế độ đồ họa bằng

```
initgraph ( &graphdriver, &graphmode, " " );
trong đó graphdriver là một biến nguyên chứa giá trị số của thiết bị màn hình hoặc chứa hàng DETECT, graphmode là một biến nguyên. Khi graphdriver chứa giá trị số của thiết bị màn hình thì graphmode chứa một trong các mốt đồ họa tương ứng với màn hình này. Khi nội dung của graphdriver bằng DETECT thì không cần khởi đầu cho graphmode, " " là tên của thư mục chứa bộ chương trình đồ họa.
```

Để chọn màu nền ta dùng hàm setbkcolor (color). Để chọn màu đường vẽ ta dùng hàm setcolor (color). Để chọn kiểu tô và màu tô ta dùng hàm setfillstyle (pattern, color). Trong cả ba trường hợp color là biến nguyên chứa mã của màu. Biến pattern chứa mã của màu tô. Trong phần phụ lục có giới thiệu một số hàm đồ họa thông dụng.

### Bài số 38 Vẽ đường biểu diễn hàm $y = f(x)$

Ung dung vẽ cho  $y_1 = \sin x$

$$y_2 = \frac{\sin x}{x}$$

## Bài giải Đây là hàm một biến cho dưới dạng tường

Để khởi tạo chế độ đồ họa ta dùng hàm intKhoiTaoDoHoa ( void ) trong đó có dùng câu lệnh if để xác định loại màn hình sử dụng có phải thuộc loại VGA hoặc CGA không. Dùng setbkcolor (CYAN) để đặt màu nền và mở cửa sổ màn hình bằng setviewport (0, getmaxy/2, get maxx(), getmaxy(), 0); cho phép vẽ hình cả ngoài cửa sổ. Vẽ trục x, trục y, viết tên trục x và y cũng như gốc (0,0)

Dùng settextstyle để định kiểu chữ cho dòng DO THI Y = f(X)

Để vẽ đồ thị hàm  $y = f(x)$  cần phải cho x biến thiên một lượng  $\Delta x$  và tính giá trị hàm y tương ứng. Để làm điều này chỉ cần một chu trình for để biến đổi x,y. Kết quả ta được một dãy trị số x,y. Để vẽ từng điểm ta dùng hàm putpixel (x,y,color). Ở đây x,y là cặp giá trị x,y nguyên và color là màu đường muôn ve.

```
/* CT38 Chương trình vẽ hàm y = f(x) */
```

```
#include <stdlib.h>
#include <math.h>
#include <graphics.h>
#define pi 3.14159255
#define TileX 20
#define TileY 100
int i,x,y;
float t;
int KhoiTaoDoHoa(void);

main()
{
    KhoiTaoDoHoa();
    setbkcolor(CYAN);
```

```

/* Mo cua so va cho phep ve hinh ca ngoai cua so */
setviewport(0,getmaxy()/2,getmaxx(),getmaxy(),0);
setcolor(LIGHTRED);
outtextxy(getmaxx()/2,10,"(0,0)");
setcolor(RED);
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve
truc Y */
linerel(0,-(getmaxy()/2-20));
outtextxy((getmaxx()/2-5),-(getmaxy()/2-20),"y");
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve
truc Y */
linerel(0,getmaxy()/2-60);
moveto(0,0); /* ve truc X */
linerel(getmaxx()-10,0);
outtextxy(getmaxx()-10,0,"x");
setcolor(BLUE);
settextstyle(TRIPLEX_FONT,HORIZ_DIR,3);
outtextxy(200,200,"DO THI Y = F(X)");
for (i=1;i<=2000;i++) /* Ham y = sin(x) */
{
    x = (int)(2*pi*i*TileX/400);
    y = - (int)(sin((2*i*pi)/400)*TileY);
    putpixel(x,y,YELLOW);
}
getch();
for (i=1;i<=2000;i++) /* Ham y =sin(x)/x */
{
    t = pi*i;
}

```

```

x = (int) (t*TileX/400+320);
y = - (int)(2*sin((t)/400)*TileY/t*200);
putpixel(x,y,WHITE);
}

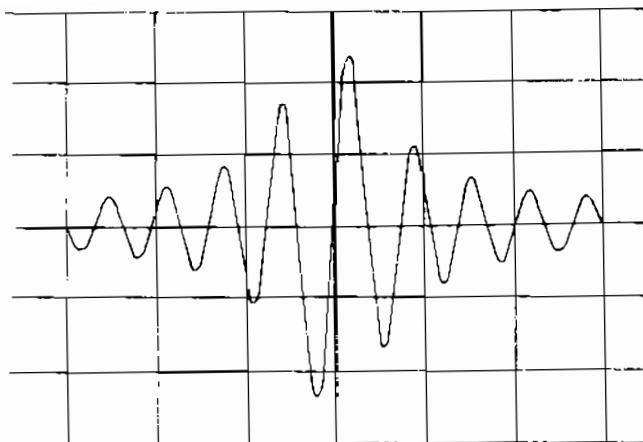
getch();
closegraph();
return;
}

int KhoiTaoDoHoa(void)
{
int graphdriver = DETECT, graphmode, error_code;
initgraph(&graphdriver, &graphmode, "..\\bgi");
error_code = graphresult();
if (error_code!= grOk)
return(-1); /* Khong phai man EGA hoac VGA */
if ((graphdriver!= EGA) && (graphdriver!= VGA))
{
closegraph();
return 0;
}
return(1);
}

```

Để vẽ hàm  $y = \sin x$  ta chọn tỷ lệ xích của  $x$  là 20, của  $y$  là 100.

Hàm  $y = \frac{\sin x}{x}$  với  $x \in [0,20]$  có dạng biểu diễn trên hình 3-2.



$$Hình 3-2 \quad y = \frac{\sin x}{x}$$

**Bài số 39** Vẽ đường biểu diễn hàm xác định theo tham số

$$\begin{cases} x = f(t) \\ y = g(t) \end{cases}$$

Ứng dụng vẽ đồ thị hàm hypocycloide  $x = \cos^3(t)$   
 $y = \sin^3(t), t \in [0, 2\pi]$

hàm trochoide  $x = 2t - 3\sin(t)$

$$y = 2 - 3\cos(t), t \in [-9, 9] \text{ và}$$

hàm  $x = (1 + \cos^2(t)) \cdot \sin(t)$

$$y = \sin^2(t) \cdot \cos(t), t \in [0, 2\pi]$$

**Bài giải** Các đường cong theo tham số được xác định bằng hai phương trình. Bằng cách biến thiên tham số  $t$  ta được dãy các cặp  $(x, y)$  tương ứng rồi dùng hàm putpixel  $(x, y, \text{color})$ ; để vẽ liên tiếp các điểm. Ta nhận thấy hàm  $y = f(x)$  là một trường hợp riêng của các hàm thông số khi lấy  $x = t$  ta được :

$$\begin{cases} x = t \\ y = g(t) = g(x) \end{cases}$$

```

/* CT39 Chuong trinh ve duong cong theo tham so */

#include <stdlib.h>
#include <math.h>
#include <graphics.h>
int i,x,y;
int KhoiTaoDoHoa(void);

main()
{
float t;
KhoiTaoDoHoa();
setbkcolor(CYAN);
/* Mo cua so va cho phep ve hinh ca ngoai cua so */
setviewport(0,getmaxy()/2,getmaxx(),getmaxy(),0);
setcolor(LIGHTRED);
outtextxy(getmaxx()/2,10,"(0,0)");
setcolor(RED);
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,-(getmaxy()/2-20));
outtextxy((getmaxx()/2-5),-(getmaxy()/2-20),"y");
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,getmaxy()/2-60);
moveto(0,0); /* ve truc X */
linerel(getmaxx()-10,0);
outtextxy(getmaxx()-10,0,"x");
setcolor(BLUE);
settextstyle(TRIPLEX_FONT,HORIZ_DIR,3);
outtextxy(200,200,"DO THI HAM THEO THAM SO");

```

```

for (i=0;i<=1000;i++)
    /* Duong Hypocycloide */
{
    t = 0.00628*i;
    x = (int)(cos(t)*cos(t)*cos(t)*120+320);
    y = (int)(sin(t)*sin(t)*sin(t)*120);
    putpixel(x,y,YELLOW);
}
getch();
for (i=-1000;i<=1000;i++)
    /* Duong Trocoide */
{
    t= 0.01*i;
    x = (int)((2*t-3*sin(t))*15+320);
    y = -(int) ((2-3*cos(t))*10);
    putpixel(x,y,WHITE);
}
getch();
for(i=0;i<=1000;i++)
    /* Duong so 3 */
{
    t= 0.00628*i;
    x = (int)((1+cos(t)*cos(t))*sin(t)*150+320);
    y = (int)(sin(t)*sin(t)*cos(t)*200);
    putpixel(x,y,RED);
}
getch();
closegraph();
return;
}

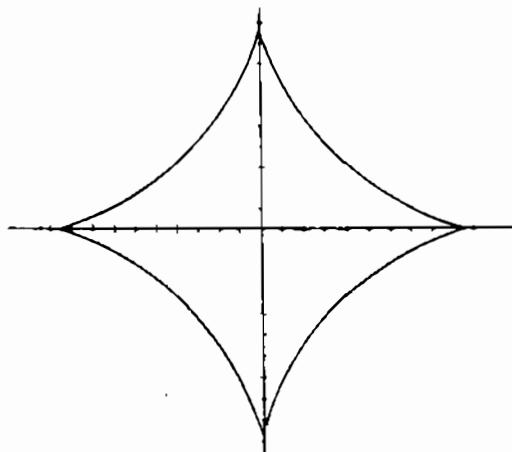
```

```

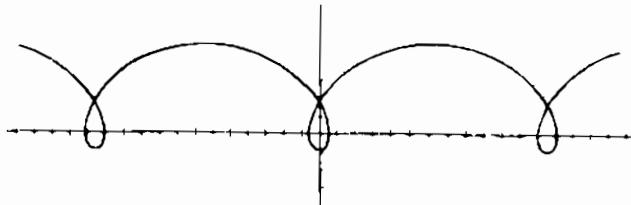
int KhoiTaoDoHoa(void)
{
    int graphdriver = DETECT, graphmode, error_code;
    initgraph(&graphdriver, &graphmode, ".\bgi");
    error_code = graphresult();
    if (error_code!= grOk)
        return(-1); /* Khong phai man EGA hoac VGA */
    if ((graphdriver!= EGA) && (graphdriver!= VGA))
    {
        closegraph();
        return 0;
    }
    return(1);
}

```

Kết quả ta được các đường biểu diễn sau đây (hình 3-3.a, b và c).

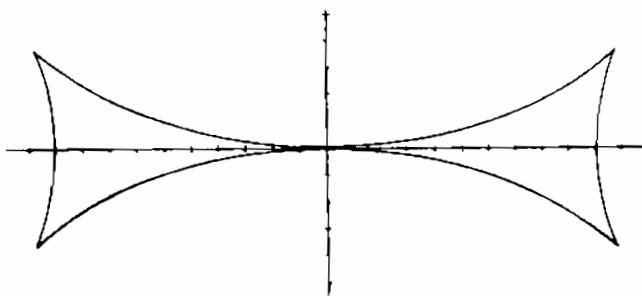


Hình 3-3 a Đường hypocycloide  $x = \cos^3(t)$   
 $y = \sin^3(t)$



Hình 3-3 b Đường trochode  $x = 2t - 3\sin(t)$

$$y = 2 - 3\cos(t)$$



Hình 3-3 c Đường  $x = (1 + \cos 2t)\sin(t)$

$$y = \sin 2t \cdot \cos t$$

#### Bài số 40 Biểu diễn đường cong theo tọa độ cực $r = f(\theta)$

Ứng dụng vẽ đường xoắn ốc Archimede  $r = \theta$ ,  $\theta \in [0, 40]$   
và đường hoa hồng bốn cánh  $r = \sin 2\theta$ ,  $\theta \in [0, 2\pi]$

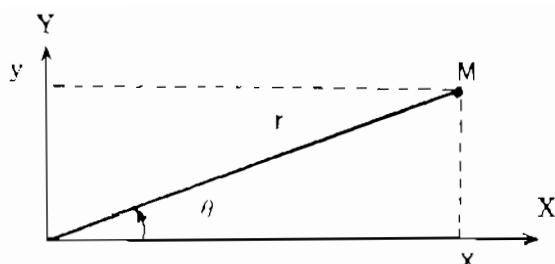
Tọa độ cực  $r, \theta$  và tọa độ xoy được biểu diễn trên hình 3-4.

Ta có quan hệ  $x = r\cos\theta$

$$y = r\sin\theta$$

và trả về việc vẽ đồ thị hàm số theo tham số. Ta dùng chu trình for để biến đổi góc  $\theta$  kết quả thu được các cặp x,y ứng với mỗi giá trị của  $\theta$ .

Đồ thị thu được vẽ trên hình 3-5 a và b



Hình 3-5

/\* CT40 Chuong trinh do hoa theo toa do cuc \*/

```
#include <stdlib.h>
#include <math.h>
#include <graphics.h>
int i,x,y;
int KhoiTaoDoHoa(void);

main()
{
    float r, theta;
    KhoiTaoDoHoa();
    setbkcolor(WHITE);
    /* Mo cua so va cho phep ve hinh ca ngoai cua so */
    setviewport(0,getmaxy()/2,getmaxx(),getmaxy(),0);
    setcolor(LIGHTRED);
    outtextxy(getmaxx()/2,10,"(0,0)");
    setcolor(RED);
    moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
    linerel(0,-(getmaxy()/2)-20);
```

```

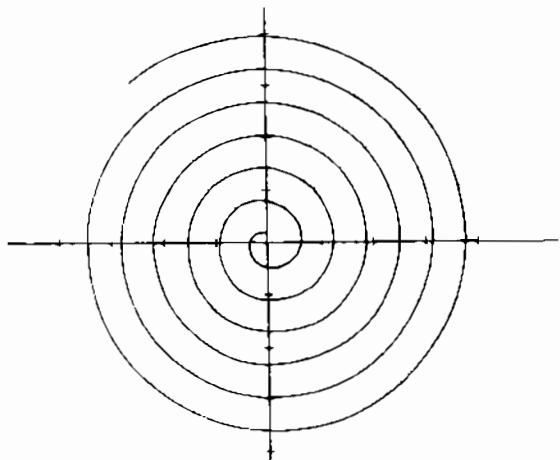
outtextxy((getmaxx()/2-5),-(getmaxy()/2-20),"y");
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,getmaxy()/2-60);
moveto(0,0),           /* ve truc X */
linerel(getmaxx()-10,0);
outtextxy(getmaxx()-10,0,"x");
setcolor(BLUE);
settextstyle(TRIPLEX_FONT,HORIZ_DIR,3);
outtextxy(200,200,"DUONG CONG THEO TOA DO CUC ");
for (i=0;i<=2000;i++)
{ /* Duong xoan oc Archimede */
    teta = 0.02*i;
    x = (int)(teta*cos(teta)*3+320);
    y = -(int)(teta*sin(teta)*3);
    putpixel(x,y,CYAN);
}
getch();
for(i=0;i<=2000;i++) /* Duong hoa hong 4 canh */
{
    teta=0.00628*i;
    x = (int)(sin(2*teta)*cos(teta)*150+320);
    y = (int)(sin(2*teta)*sin(teta)*150);
    putpixel(x,y,RED);
}
getch();
closegraph();
return;
}

```

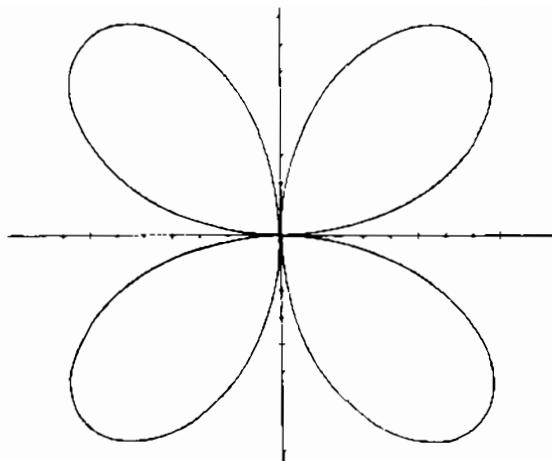
```

int KhoiTaoDoHoa(void)
{
    int graphdriver = DETECT, graphmode, error_code;
    initgraph(&graphdriver, &graphmode, "Abgi");
    error_code = graphresult();
    if (error_code != grOk)
        return(-1); /* Khong phai man EGA hoac VGA */
    if ((graphdriver != EGA) && (graphdriver != VGA))
    {
        closegraph();
        return 0;
    }
    return(1);
}

```



Hình 3-5 a  $r=\theta$  đường xoắn ốc Archimede



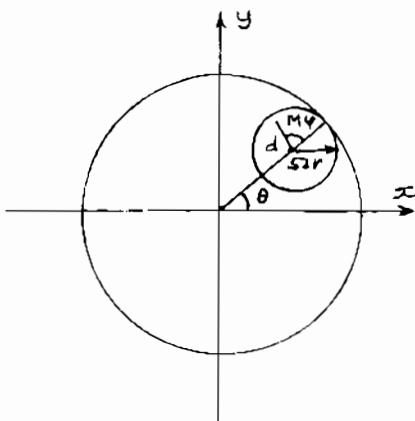
Hình 3-5 b  $r = \sin 2\theta$ , đường hoa hồng bốn cánh

#### Bài số 41 Vẽ đường cong cycloide

Một đĩa tâm  $\Omega$  bán kính  $r$  quay bên trong và tiếp xúc với vòng tròn tâm  $O$  bán kính  $R$ .

Điểm  $M$  trên đĩa mô tả đường cong cycloide (hình 3-6)

Đường cycloide là trường hợp riêng của đường cong theo tham số. Gọi  $M$  là vị trí của điểm cần tìm quỹ đạo,  $O$  là tâm đường tròn,  $\Omega$  là tâm của đĩa. Gọi  $d = \Omega M$  là khoảng cách giữa  $M$  và tâm đĩa và đặt  $d = k \frac{r}{10}$ . Như vậy khi  $k$  biến thiên giữa 0 và 10 điểm  $M$  chạy từ tâm vòng tròn đến chu vi. Khi  $k$  càng gần 10 đường cong càng nhọn ngược lại  $k$  càng gần 0



Hình 3-6

đường cong còng tròn. Ta có :

$$OM = O\Omega + \Omega M$$

$$x = (R - r) \cos\theta + d \cos(\theta + \varphi)$$

$$y = (R - r) \sin\theta + d \sin(\theta + \varphi)$$

Vì đĩa quay trong vòng tròn ta có  $r\varphi = -R\theta$ . Khử  $\varphi$  ta được

$$x = (R - r)\cos\theta + \frac{kr}{10} \cos\left[\left(1 - \frac{R}{r}\right)\theta\right]$$

$$y = (R - r)\sin\theta + \frac{kr}{10} \sin\left[\left(1 - \frac{R}{r}\right)\theta\right]$$

Chọn  $R = 1$ , đặt  $r = \frac{n}{N}$  ta có :

$$x = \left(1 - \frac{n}{N}\right)\cos\theta + k \frac{n}{10N} \cos\left[\left(1 - \frac{N}{n}\right)\theta\right]$$

$$y = \left(1 - \frac{n}{N}\right)\sin\theta + k \frac{n}{10N} \sin\left[\left(1 - \frac{N}{n}\right)\theta\right]$$

Đây là phương trình tham số theo  $\theta$ . Cho  $\theta$  biến thiên từ 0 đến  $2\pi$ , phụ thuộc vào trị số của  $n$ ,  $N$  và  $k$  ta được các quỹ đạo cycloide khác nhau (hình 3-7)

```
/* CT41 Chuong trinh ve quy dao Cycloide */
```

```
#include <stdlib.h>
#include <math.h>
#include <graphics.h>
int i,x,y;
int KhoiTaoDoHoa(void);
```

```
main()
```

```
{
```

```
int n,N,k;
```

```
float teta;
```

```
printf("\n Vao du lieu ");
```

```

printf("\n Cho n = "); scanf("%d",&n);
printf("\n Cho N = "); scanf("%d",&N);
printf("\n Cho k = "); scanf("%d",&k);
KhoiTaoDoHoa();
setbkcolor(CYAN);
/* Mo cua so va cho phep ve hinh ca ngoai cua so */
setviewport(0, getmaxy()/2, getmaxx(),getmaxy(),0);
setcolor(LIGHTRED);
outtextxy(getmaxx()/2,10,"(0.0)");
setcolor(RED);
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,-(getmaxy()/2-20));
outtextxy((getmaxx()/2-5), -(getmaxy()/2-20),"y");
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,getmaxy()/2-60);
moveto(00); /* ve truc X */
linerel(getmaxx()-10,0);
outtextxy(getmaxx()-10,0"x");
setcolor(BLUE);
settextstyle(TRIPLEX_FONT,HORIZ_DIR,3);
outtextxy(200,200,"DO THI QUY DAO");
for (i= 0;i<=1000;i++)
{
    teta = 0.00628*i;
    x = (int)((1-n/N)*cos(teta)+k*n*cos((1-N/n)*teta)/(10*N))
        *150 + 320);
    y = (int)((1-n/N)*sin(teta)+k*n*sin((1-N/n)*teta)/(10*N))*150;
}

```

```

    putpixel(x,y,YELLOW);
}

getch();
closegraph();
return;
}

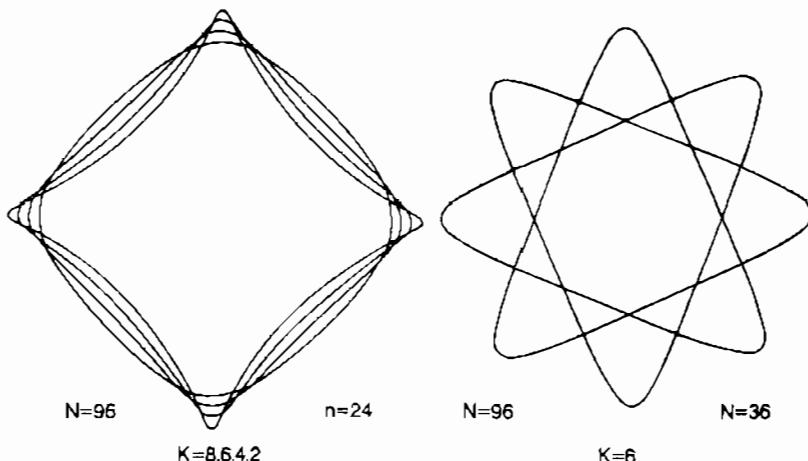
int KhoiTaoDoHoa(void)
{
    int graphdriver = DETECT, graphmode, error_code;
    initgraph(&graphdriver, &graphmode, "..\bgi");
    error_code = graphresult();
    if (error_code != grOk)
        return(-1); /* Khong phai man EGA hoac VGA */
    if ((graphdriver != EGA) && (graphdriver != VGA))
    {
        closegraph();
        return 0;
    }
    return(1);
}

```

**Bài số 42** Vẽ mặt trong không gian ba chiều hàm hai biến  $z = f(x,y)$ .

Trường hợp cụ thể  $z = 5 \sin x \sin y$

**Bài giải** Ta cần tìm cách biểu diễn không gian lên mặt phẳng. Vì màn hình chỉ có hai chiều, để vẽ hình không gian ba chiều ta phải chiếu không gian lên mặt phẳng. Có nhiều phương pháp chiếu, ở đây ta sử dụng phép chiếu song song. Màn hình có tọa độ OXY và tọa độ không gian oxyz (hình 3-8) Oz là trục thẳng đứng, góc giữa oX và ox là  $\alpha$ ; góc



Hình 3-7

$oX$  và  $oy$  là  $\beta$ . Điểm  $O (X_o, Y_o)$  có tọa độ màn hình là :

$$Oo = X_o \vec{i} + Y_o \vec{j}$$

$$\vec{i} = \cos(\alpha) \vec{I} + \sin(\alpha) \vec{J}$$

$$\vec{j} = \cos(\beta) \vec{I} + \sin(\beta) \vec{J}$$

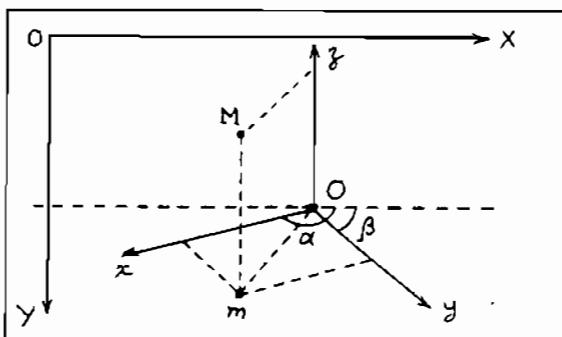
$$\vec{k} = -\vec{J}$$

Gọi  $M$  là một điểm trong không gian các tọa độ là các số thực  $(x, y, z)$ .

Chiếu lên mặt phẳng màn hình có tọa độ  $(X, Y)$

$$oM = x \vec{i} + y \vec{j} + z \vec{k}$$

$$OM = X \vec{i} + Y \vec{j}$$



Hình 3-8

Ta được các công thức chuyển từ tọa độ không gian về tọa độ màn hình :

$$X = X_0 + x \cos(\alpha) + y \cos(\beta)$$

$$Y = Y_0 + x \sin(\alpha) + y \sin(\beta) - z$$

Trong hệ thống ta xác định một điểm trong không gian bằng 2 số thực và 5 số nguyên :

- hai số nguyên cho tọa độ gốc O trong OXY
- ba số nguyên tính bằng pixel cho các đơn vị trên trục ox,oy,oz
- hai số thực chỉ góc giữa OX và các trục ox,oy. Trong chương trình tọa độ màn hình được định nghĩa bằng cấu trúc `gr3_toado`.

Để chuyển tọa độ cần xây dựng hai hàm :

```
int X (double x, double y, double z)
```

```
int Y (double x, double y, double z) theo các công thức chuyển  
tọa độ.
```

Hàm cần vẽ được định nghĩa bằng hàm hai biến dạng :

```
double ham ( double x, double y );
```

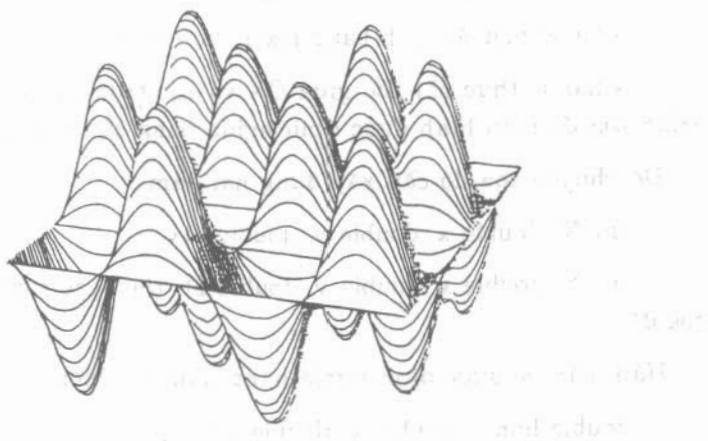
Mặt được biểu diễn bằng dây liên tiếp các đường cong cắt bằng các mặt phẳng song song theo mặt xoy hoặc yoz tùy theo vị trí của người quan sát. Mỗi đường cong được tạo nên bởi dây các đoạn thẳng nối với các điểm tính toán trên mặt. Đa số mặt với độ chính xác chấp nhận cần vẽ khoảng 50 đường cong và số điểm tính toán khoảng 100.

Để mặt biểu diễn dễ quan sát nên loại bỏ các đường khuất theo nguyên lý sau đây :

Theo đường vẽ ta xác định hai đường biên biểu diễn các điểm cực đại và cực tiểu do các đường cong khác nhau tạo ra trên mặt. Khi vẽ đường mới vị trí của các điểm của đoạn đang vẽ được so sánh với vùng khuất bao gồm giữa hai đường biên.

Trong chương trình phần khởi tạo chế độ đồ họa, xây dựng cấu trúc tọa độ màn hình struct gr3\_toado, xây dựng hàm chuyển tọa độ X và Y, xây dựng hàm  $z = \text{ham}(x,y)$  tương tự bài trên.

Để vẽ có loại bô đường khuất ta tạo nên một mảng gọi tên là mat[8] để chứa tọa độ X,Y để kiểm tra điều kiện tô màu và dùng fillpoly (4,mat) ; để vẽ và tô màu đa giác có đỉnh là các điểm  $(x_1,y_1) \dots (x_4,y_4)$ . Kết quả ta được hình biểu diễn của hàm  $z = \sin xy$  (hình 3-9).



Hình 3-9

```
/* CT42 Chuong trinh ve mat khong gian 3 chieu mat an */
#include <stdio.h>
#include <math.h>
#include <graphics.h>
typedef struct /* toa do man hinh */
{
    int OX, /* truc hoanh man hinh */
    OY, /* truc tung man hinh */
    UX, /* don vi truc hoanh */
```

```

UY,      /* don vi truc tung */
UZ;      /* don vi truc oz */
double Xx, /* goc (OX,ox) */
Xy ;     /* goc (OX,oy) */
} gr3_toado;
gr3_toado gr3 = { 250,20,20,20,20,0.8*M_PI,0.2*M_PI};
int gr3_cla = GREEN,    /* mau cac truc */
gr3_clt = YELLOW,    /* mau duong */
gr3_clf = BLUE,     /* mau nen */
gr3_clc = RED;      /* mat an */
double gr3_lx = 2,
gr3_ly = 2,
gr3_lz = 2;
int X(double x,double y,double z); /* chuyen toa do */
int Y(double x,double y,double z);
double ham (double,double);        /* ham can ve */
int KhoiTaoDoHoa(void);

main()
{
double x,y,z, xa=-6.28,xb=6.28,ya=-6.28,yb=6.28,xp=0.2,yp=0.2;
int mat[8];
KhoiTaoDoHoa();
setbkcolor(CYAN);
/* Mo cua so va cho phep ve hinh ca ngoai cua so */
setviewport(0,getmaxy()/2,getmaxx(),getmaxy(),0);
setcolor(LIGHTRED);
outtextxy(getmaxx()/2,10,"(0,0)");

```

```

setcolor(RED);
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,-(getmaxy()/2-20));
outtextxy((getmaxx()/2-5),-(getmaxy()/2-20),"y");
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,getmaxy()/2-60);
moveto(0,0); /* ve truc X */
linerel(getmaxx()-10,0);
outtextxy(getmaxx()-10,0,"x");
setcolor(BLUE);
settextstyle(TRIPLEX_FONT,HORIZ_DIR,3);
outtextxy(200,-220,"DO THI KHONG GIAN 3 CHIEU");
for (x=xa;x<=xb;x+=xp) /* Ve co the hien mat an */
  for(y=ya ;y<=yb ;y+=yp)
  {
    if( kbhit() ) return;
    z= ham(x,y);           /* diem A */
    mat[0] = X(x,y,z);
    mat[1] = Y(x,y,z);
    z =ham(x,y+yp);       /* diem B */
    mat[2] = X(x,y+yp,z);
    mat[3] = Y(x,y+yp,z);
    z = ham(x+xp,y+yp); /* diem C */
    mat[4] = X(x+xp,y+yp,z);
    mat[5] = Y(x+xp,y+yp,z);
    z = ham(x+xp,y);     /* diem D */
    mat[6] = X(x+xp,y,z) ;
    mat[7] = X(x+xp,y,z) ;
  }

```

```

if ( (mat[3]-mat[1])*(mat[6]-mat[0])-  

    (mat[7]-mat[1])*(mat[2]-mat[0])<0)  

    setfillstyle(1,YELLOW);  

else  

    setfillstyle(1,GREEN);  

fillpoly(4,mat);  

}  

closegraph();  

return;  

}  

int X(double x,double y,double z) /* Ham chuyen toa do */  

{  

return( (gr3.OX+  

    x*gr3.UX*cos(gr3.Xx)+  

    y*gr3.UY*cos(gr3.Xy)));  

}  

int Y(double x,double y,double z)  

{  

return( (gr3.OY+  

    x*gr3.UX*sin(gr3.Xx)+  

    y*gr3.UY*sin(gr3.Xy)-z*gr3.UZ));  

}  

double ham (double x, double y) /* Ham can ve */  

{  

return( 5*sin(x)*sin(y));  

}

```

```

int KhoiTaoDoHoa(void) /* Ham khoi tao che do do hoa */
{
    int graphdriver = DETECT, graphmode, error_code;
    initgraph(&graphdriver, &graphmode, ".\\bgi");
    error_code = graphresult();
    if (error_code != grOk)
        return(-1); /* Khong phai man EGA hoac VGA */
    if ((graphdriver != EGA) && (graphdriver != VGA))
    {
        closegraph();
        return 0;
    }
    return(1);
}

```

**Bài số 43** Vẽ mặt trong không gian ba chiều hàm hai biến  $z = f(x,y)$

$$\text{Trường hợp cụ thể } z = \frac{5\sin x^2 + y^2}{x^2 + y^2}$$

**Bài giải** Tương tự bài số 42 ở đây ta cũng sử dụng phép chiếu không gian lên mặt phẳng và dùng các công thức chuyển đổi từ tọa độ không gian về tọa độ màn hình.

Chương trình CT43 gồm các bước sau :

- Xác định màn hình bằng cấu trúc gr3\_toado
- Xây dựng hai hàm chuyển tọa độ :

```
int X (double x,double y, double z)
```

```
int Y (double x,double y, double z)
```

- Xác định hàm cần vẽ đồ thị bằng :

double ham ( double x, double y ) ;

- Cho x biến thiên từ -10 đến +10 với bước  $x_p = 0,1$

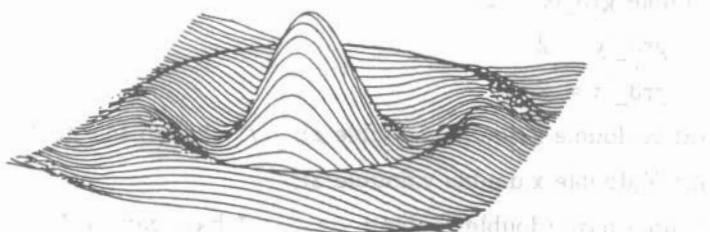
Cho y biến thiên từ -10 đến +10 với bước  $y_p = 0,1$

gọi hàm z = ham (x,y);

và dùng lineto ( X(x,y,z), Y(x,y,z));

để vẽ đường thẳng từ điểm hiện tại tới điểm (X,Y) và chuyển con chạy tới vị trí điểm (X,Y). Kết quả ta được mặt biểu diễn hàm hai biến

$$z = \frac{5 \sin \sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2}}$$
 bằng phép chiếu song song (hình 3-10)



Hình 3-10

/\* CT43 Chuong trinh ve mat khong gian 3 chieu \*/

#include <stdio.h>

#include <math.h>

#include <graphics.h>

```

typedef struct      /* toa do man hinh */
{
    int OX,          /* truc hoanh man hinh */
    UX,          /* truc tung man hinh */
    UY,          /* don vi truc hoanh */
    UZ;          /* don vi truc oz */
    double Xx,        /* goc (OX,ox) */
    Xy;          /* goc (OX,oy) */
} gr3_toado;

gr3_toado gr3 = { 320,25,20,20,20,0.8*M_PI,0.2*M_PI};

int gr3_cla = GREEN,     /* mau cac truc */
    gr3_clt = YELLOW,    /* mau duong */,
    gr3_clf = BLUE,       /* mau nen */
    gr3_cle = RED;        /* mat an */

double gr3_lx = 2,
    gr3_ly = 2,
    gr3_lz = 2;

int X(double x,double y,double z); /* chuyen toa do */
int Y(double x,double y,double z);
double ham (double,double);        /* ham can ve */
int KhoiTaoDoHoa(void);

main()
{
    double x,y,z, xa=-10,xb=10,ya=-10,yb=10,xp=0.1,yp=0.1;
    int mat[8];
    KhoiTaoDoHoa();
    setbkcolor(CYAN);
}

```

```

/* Mo cua so va cho phep ve hinh ca ngoai cua so */
setviewport(0,getmaxy()/2,getmaxx(),getmaxy(),0);
setcolor(LIGHTRED);
outtextxy(getmaxx()/2,10,"(0,0)");
setcolor(RED) ;
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,-(getmaxy()/2-20));
outtextxy((getmaxx()/2-5),(getmaxy()/2-20),"y");
moveto(getmaxx()/2-5,10); /* Chuyen con tro ve goc va ve truc Y */
linerel(0,getmaxy()/2-60);
moveto(0,0); /* ve truc X */
linerel(getmaxx()-10,0);
outtextxy(getmaxx()-10,0,"x");
setcolor(YELLOW);
settextstyle(TRIPLEX_FONT,HORIZ_DIR,3);
outtextxy(200,-240,"DO THI KHONG GIAN 3 CHIEU");
for (x=xa;x<=xb;x+=xp) /* Ve do thi */
{
    for(y=ya;y<=yb;y+=yp)
    {
        if( kbhit() ) return;
        z= ham(x,y);
        lineto(X(x,y,z), Y(x,y,z));
    }
    closegraph();
    return;
}
int X(double x,double y,double z) /* Ham doi toa do */
| return( gr3.OX+

```

```

x*gr3.UX*cos(gr3.Xx) +
y*gr3.UY*cos(gr3.Xy)));
}

int Y(double x,double y,double z)
{ return( (gr3.OY+
x*gr3.UX*sin(gr3.Xx) +
y*gr3.UY*sin(gr3.Xy)-z*gr3.UZ));
}

double ham (double x, double y) /* Ham can ve do thi */
{
return( 5*sin(sqrt(x*x+y*y))/sqrt(x*x+y*y));
}

int KhoiTaoDoHoa(void) /* Ham khoi tao che do do hoa */
{
int graphdriver = DETECT, graphmode, error_code;
initgraph(&graphdriver, &graphmode, "..\\bgi");
error_code = graphresult();
if (error_code!= grOk)
    return(-1); /* Khong phai man EGA hoac VGA */
if ((graphdriver!= EGA) && (graphdriver!= VGA))
{
closegraph();
return 0;
}
return(1);
}

```

#### Bài số 44 Vẽ biểu đồ thống kê các dữ liệu.

**Bài giải** Chương trình chính gồm việc nhập các dữ liệu, tính giá trị phần trăm của dữ liệu và in kết quả theo giá trị và theo phần trăm. Tiếp theo là vẽ biểu đồ dạng hình hộp.

Dùng hàm bar3d để vẽ các khối hộp chữ nhật có kích thước chiều cao phụ thuộc vào giá trị thống kê. Số lượng khối hình chữ nhật bằng số giá trị thống kê, ở đây số giá trị thống kê cực đại chọn là 6. Kết quả cho trên hình 3-11.

```
/* CT44 Chuong trinh ve bieu do thong ke */
#include <stdio.h>
#include <stdlib.h>
#include <float.h>
#include <graphics.h>
#include <math.h>
#include <conio.h>
#define MAX 100
#define SoGiaTri 6
void VeBieuDo(float p[]);      /* Ham ve bieu do */
void main(void)
{
    int i;
    int giatri[SoGiaTri];
    float phantram[SoGiaTri];
    for (i = 0; i < SoGiaTri; i++)
    {
        printf("\nCho gia tri thong ke (giua 0 va %d) : ",MAX);
        scanf("%d", &giatri[i]);
    }
}
```

```

for (i = 0; i < SoGiaTri; i++)
    phantram[i] = ((float) giatri[i]) / MAX;
printf("\n\n\nNo tGIA TRI t;tSo %");
for (i = 0;i<SoGiaTri;i++)
    printf("\n%d.\t%d\t%3.0f",i+1,giatri[i],(phantram[i]*100));
getch();
VeBieuDo(phantram);
}

void VeBieuDo(float p[])
{
int gd, gm;
int i, trai, tren, rong, duoi, sau;
detectgraph(&gd,&gm);
initgraph(&gd,&gm,"");
setbkcolor(GREEN);
setcolor(RED);
setTextStyle(TRIPLEX_FONT,HORIZ_DIR,3);
outtextxy(200,30,"BIEU DO THONG KE");
rong = (int)((getmaxx()) / ((SoGiaTri * 2 ) + 1));
duoi = getmaxy() - 20;
sau = (int) (rong / 4);
trai = rong;
for (i = 0; i < SoGiaTri; i++)
{
    tren = (duoi) - ((int)(p[i] * 300));
    bar3d(trai,tren,(trai + rong),duoi,sau,1);
    trai += (rong * 2);
}

```

```

getch();
closegraph();
return;
}

```

*Cho gia tri thong ke ( giua 0 va 100 ) : 48*

*Cho gia tri thong ke ( giua 0 va 100 ) : 35*

*Cho gia tri thong ke ( giua 0 va 100 ) : 29*

*Cho gia tri thong ke ( giua 0 va 100 ) : 71*

*Cho gia tri thong ke ( giua 0 va 100 ) : 15*

*Cho gia tri thong ke ( giua 0 va 100 ) : 38*

No	GIA TRI	So%
1.	46	46
2.	35	35
3.	29	29
4.	71	71
5.	15	15
6.	38	38



Hinh 3-11

### Bài số 45 Vẽ hệ mặt trời

**Bài giải** Hệ mặt trời gồm các hành tinh, tinh từ vị trí gần mặt trời nhất là :

Mercury ( sao Thủy ) Venus ( sao Kim ) Earth ( Trái đất ) Mars ( sao Hỏa ) Jupiter ( sao Mộc ) Saturn ( sao Thổ ) Uranus ( sao Hải vương ).

Chương trình chính ngoài hàm khởi tạo chế độ đồ họa còn có :

- Hàm tính tọa độ hành tinh void TinhToaDo(void), xác định tọa độ của các hành tinh dùng các biến nguyên au1 là đơn vị thiên văn tính bằng pixel cho hành tinh nhỏ hơn trái đất.

au2- đơn vị thiên văn tính bằng pixel cho các hành tinh lớn hơn trái đất.

erad - bán kính trái đất tính bằng pixel.

- Hàm vẽ hành tinh , vẽ vòng tròn có tọa độ và bán kính xác định theo tỷ lệ so với bán kính trái đất. Mặt trời được biểu diễn bằng một cung tròn.

Hình vẽ hệ mặt trời cho trên hình 3-12.



Hình 3-12

```
/* CT45 Chuong trinh ve he Mat troi */  
#include <graphics.h>  
#include <stdlib.h> /* Dung cho lenh exit() */
```

```

#include <stdio.h>
#include <conio.h>
int KhoiTaoDoHoa(void);
void TinhToaDo(void); /* Ty le khoang cach tren man hinh */
void VeCacHanhTinh(void);
void VeHanhTinh(float ToaDo_x, float Bankinh,
                int Mau, int Kieu_to);
void get_key(void); /* wait for key */
/* Bien tong the cho ham TinhToaDo */
int max_x, max_y; /* Gia tri Max cua truc x va truc y */
int y_org; /* Truc y cho toan hinh ve */
int au1; /* Don vi thien van tinh bang pixels
            cho hanh tinh nho hon Trai dat */
int au2; /* Don vi thien van tinh bang pixels
            cho hanh tinh lon hon Trai dat */
int erad; /* Ban kinh Trai dat bang pixels */

int main()
{
/* Kiem tra xem man hinh la EGA hoac VGA */
if (KhoiTaoDoHoa() != 1)
{
    printf("\nChuong trinh nay dung cho man hinh EGA
           hoac VGA\n");
    exit(0);
}
TinhToaDo(); /* Thang ti le theo do phan giao man hinh */
VeCacHanhTinh();
get_key();

```

```

closegraph();
return 0;
}

int KhoiTaoDoHoa(void)
{
int graphdriver = DETECT, graphmode, error_code;
initgraph(&graphdriver, &graphmode, ".\bgi");
error_code = graphresult();
if (error_code!= grOk)
    return(-1);      /* Khong phai man EGA hoac VGA */
if ((graphdriver!= EGA) && (graphdriver!= VGA))
{
    closegraph();
    return 0;
}
return(1);
}

void TinhToaDo(void)
{
max_x = getmaxx();
max_y = getmaxy();
y_org = max_y / 2;      /* Toa do Y cho moi vat the */
erad = max_x / 200;     /* Ban kinh Trai dat tinh bang pixels */
au1 = erad * 20;        /* Ti le cho hanh tinh nho hon Trai dat */
au2 = erad * 10;        /* Ti le cho hanh tinh lon hon Trai dat */
}

```

```

void VeCacHanhTinh()
{
    /* Cung Mat troi */
    VeHanhTinh(-90, 100, YELLOW, EMPTY_FILL);

    /* Mercury */
    VeHanhTinh(0.4 * au1, 0.4 * erad, BROWN, LTBKSLASH_FILL);

    /* Venus */
    VeHanhTinh(0.7 * au1, 1.0 * erad, WHITE, SOLID_FILL);

    /* Earth */
    VeHanhTinh(1.0 * au1, 1.0 * erad, LIGHTBLUE, SOLID_FILL);

    /* Mars */
    VeHanhTinh(1.5 * au1, 0.4 * erad, LIGHTRED,
CLOSE_DOT_FILL);

    /* Jupiter */
    VeHanhTinh(5.2 * au2, 11.2 * erad, WHITE, LINE_FILL);

    /* Saturn */
    VeHanhTinh(9.5 * au2, 9.4 * erad, LIGHTGREEN, LINE_FILL);

    /* Uranus */
    VeHanhTinh(19.2 * au2, 4.2 * erad, GREEN, LINE_FILL);
}

void VeHanhTinh(float ToaDo_x, float BanKinh, int Mau, int
Kieu_to)
{
    setcolor(Mau);
}

```

```
circle(ToaDo_x, y_org, BanKinh);
setfillstyle(Kieu_to, Mau);
floodfill(ToaDo_x, y_org, Mau);
}

void get_key(void)
{
outtextxy(50, max_y - 20, "An phim bat ky de ket thuc");
getch();
}
```

## Chương 4

# VECTƠ, MA TRẬN, HỆ PHƯƠNG TRÌNH ĐẠI SỐ TUYẾN TÍNH

**Bài số 46** Cho hai vectơ trong tọa độ ba chiều. Lập chương trình tính tổng, hiệu, tích vô hướng và tích của hai vectơ.

### Bài giải

Tổng hai vectơ có tọa độ là :  $x_1 + x_2, y_1 + y_2, z_1 + z_2$

Hiệu hai vectơ có tọa độ là :  $x_1 - x_2, y_1 - y_2, z_1 - z_2$

Tích vô hướng hai vec tơ có tọa độ là :  $x_1 * x_2, y_1 * y_2, z_1 * z_2$

Tích vectơ có tọa độ là :

$$y_1 * z_2 - z_1 * y_2, z_1 * x_2 - x_1 * z_2, x_1 * y_2 - y_1 * x_2$$

```
/* CT46 Chuong trinh tinh cac dai luong vecto */
```

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
float x1,x2,y1,y2,z1,z2;
```

```
printf("\n Chuong trinh tinh vecto tao do 3 chieu\n");
```

```
printf("Cho tao do vecto 1 :").
```

```

scanf("%f %f %f",&x1,&y1,&z1);
printf("\n");
printf("Cho toa do vecto 2 .");
scanf("%f %f %f",&x2,&y2,&z2);
printf("\n");
printf(" Vecto 1 + Vecto 2 = %6.2f %6.2f %6.2f\n",
       x1+x2,y1+y2,z1+z2);
printf(" Vecto 1 - Vecto 2 = %6.2f %6.2f %6.2f\n",
       x1-x2,y1-y2,z1-z2);
printf(" Tich vo huong = %6.2f %6.2f %6.2f\n",
       x1*x2,y1*y2,z1*z2);
printf(" Tich vecto = %6.2f %6.2f %6.2f\n",
       y1*z2-z1*y2,z1*x2-x1*z2,x1*y2-y1*x2);
getch();
return;
}

```

Kết quả chương trình CT46 là :

*Cho toa do vecto 1 : 12 34 -6.57  
 Cho toa do vecto 2 : 42.7 24 -6.25  
 Vecto 1 + vecto 2 = 54.70 58.00 -13.00  
 Vecto 1 + vecto 2 = -30.70 10.00 -0.50  
 Tich vo huong = 512.40 816.00 42.19  
 Tich vecto = -50.50 -213.23 -1163.80*

#### Bài số 47 Cho hai vecto n chiều

$$x = (x_1, x_2, \dots, x_n)$$

$$y = (y_1, y_2, \dots, y_n)$$

Tính vectơ  $z = (z_1, z_2, \dots, z_n)$  có các thành phần xác định như sau :

$$\begin{array}{ll} 0 & \text{nếu } 0 < x_i + y_i < A, A > 0. \\ z_i = x_i + y_i & \text{nếu } x_i + y_i \geq A \\ (x_i + y_i)^2 & \text{nếu } x_i + y_i \leq 0, i = 1, 2, \dots, n \end{array}$$

và tính tích vô hướng của  $x$  và  $z$

Bài giải Chương trình CT47 gồm việc nhập dữ liệu hai vectơ  $x$  và  $y$ , một số dương và thử các điều kiện  $x_i = y_i$  để rẽ nhánh chương trình.

```
/* CT47 Chuong trinh tinh tich vecto n chieu */
#include <stdio.h>
#include <math.h>
typedef float vt[100];

main()
{
    int i,n;
    float p,a;
    vt x,y,z;
    printf("\nChuong trinh tinh tich vecto n chieu\n");
    printf(" Cho chieu cua 2 vecto n = ") ;scanf("%d",&n);
    printf(" Cho toa do cua 2 vecto\n");
    for (i=1;i<=n;i++)
    {
        printf("x[%d] = ",i);
        scanf("%f",&x[i]);
        printf("y[%d] = ",i);
        scanf("%f",&y[i]));
    }
}
```

```

printf("\n Vao so duong a = ");
scanf("%f",&a);
for (i=1;i<=n;i++)
{
    if ((x[i]+y[i])<=0) z[i]=pow((x[i]+y[i]),2);
    else if ((x[i]+y[i])<a) z[i]=0;
    else if ((x[i]+y[i])>=a) z[i]=x[i]+y[i];
    printf("z[%d] = %8.2f\n",i,z[i]);
}
p=0;
for (i=1;i<=n;i++)
p+=x[i]*z[i];
printf("Tich vo huong cua x va z la : %8.2f\n",p);
getch();
return;
}

```

Kết quả ta được :

*Chuong trinh tinh tích vecto n chieu*

*Cho chieu cua 2 vecto n = 3*

*Cho toa do cua hai vecto*

*x[1] = 2.75*

*y[1] = -5.25*

*x[2] = 28.65*

*y[2] = 23.75*

*x[3] = -4*

*y[3] = 20.15*

Vao so duong a = 2

z[1] = 6.25

z[2] = 52.40

z[3] = 16.15

Tich vo huong cua x va z la : 1453.85

#### Bài số 48 Tính ma trận chuyển vị của ma trận A

Bài giải Cho ma trận A, ma trận chuyển vị có  $b_{ij} = a_{ji}$

Dùng hai chương trình for để thay đổi chỉ số hàng và cột và hoán vị chỉ số hàng cột.

```
/* CT48 Chuong trinh tinh ma tran chuyen vi */
#include <stdio.h>
typedef float mt[10][10];

main()
{
    int i,j,n,m;
    float x;
    mt a,b;

    printf("\n%20c Tinh ma tran chuyen vi cua mot ma tran \n");
    /* Cach tao blanc : bang so ky tu trong (%20c) duoc in ra */
    /* Chu y : so blanc (' ') khong co y nghia va khong duoc qua 3 */
    printf(" Cho so chieu cua ma tran\n");
    printf(" So hang m = ") ;scanf("%d",&m);
    printf(" So cot n = ") ;scanf("%d",&n);
    printf(" Vao ma tran a[m,n]\n");
```

```

for (i=1;i<=m;i++)
    for (j=1;j<=n;j++)
    {
        printf(" a[%d][%d] = ",i,j);
        scanf("%f",&x);
        a[i][j] = x;
    }
for (i=1;i<=n;i++)
    for (j=1;j<=m;j++)
        b[i][j]=a[j][i];
for (i=1;i<=m;i++)
{
    for (j=1;j<=n;j++)
        printf("a[%d][%d] = %8.2f ",i,j,a[i][j]);
    printf("\n");
}
printf("\n\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=m;j++)
        printf("b[%d][%d] = %8.2f ",i,j,b[i][j]);
    printf("\n");
}
/* Cach in kieu khac */
printf("Ma tran A\n");
for (i=1;i<=m;i++)
    for (j=1;j<=n;j++)
        printf("%8.2f%c",a[i][j],j==n?'`':',');

```

```

printf("\n\n");
printf("Ma tran chuyen vi B\n");
for (i=1;i<=n;++)
    for (j=1;j<=m;++)
        printf("%8.2f%c",b[i][j],j==m?'\\n':',');
getch();
return;
}

```

Kết quả chương trình CT48 như sau :

*Số hàng m = 3*

*Số cột n = 2*

*Vào ma trận a[m,n]*

*a[1][1] = 12*

*a[1][2] = 34*

*a[2][1] = 45*

*a[2][2] = 78*

*a[3][1] = -95*

*a[3][2] = 67*

*a[1][1] = 12.00    a[1][2] = 34.00*

*a[2][1] = 45.00    a[2][2] = 78.00*

*a[3][1] = 67.00    b[3][2] = 67.00*

*b[1][1] = 12.00    b[1][2] = 45.00    b[1][3] = -95.00*

*b[2][1] = 34.00    b[2][2] = 78.00    b[2][3] = 67.00*

*ma trận A*

*12.00    34.00*

*45.00    78.00*

*-95.00    67.00*

*Ma tran chuyen vi B*

12.00	45.00	-95.00
34.00	78.00	67.00

**Bài số 49** Tìm số nhỏ nhất trong các phần tử của một ma trận.

**Bài giải** Chương trình gồm việc vào dữ liệu ma trận m hàng n cột và dùng hai câu lệnh for để thay đổi chỉ số hàng i, cột j. Thuật toán tìm số lớn nhất max và nhỏ nhất min tương tự như thuật toán trong chương trình CT27.

```
/* CT49 Chuong trinh tim Min va Max trong ma tran */
#include <stdio.h>
float a[10][10];

main()
{
int i,j,m,n;
float max,min,x;
printf("\nChuong trinh tim MAX va MIN trong ma tran\n");
printf("Cho hang cua ma tran m = ");
scanf("%d",&m);
printf("Cho cot cua ma tran n = ");
scanf("%d",&n);
printf("\nVao ma tran A\n");
for (i=1;i<=m;i++)
{
    for (j=1;j<=n;j++)
    {
        printf("A[%d,%d] = ",i,j);
        scanf("%f",&x);
        if (x < min) min = x;
        if (x > max) max = x;
    }
}
printf("So lon nhat la %f",max);
printf("So nho nhat la %f",min);
```

```

    a[i][j]=x;
}

max = a[1][1];
min = a[1][1];
for (i=1;i<=m;i++)
    for (j=1;j<=n;j++)
    {
        if (max>a[i][j])
            max = max;
        else
            max = a[i][j];
        if (min<a[i][j])
            min = min;
        else
            min = a[i][j];
    }
printf(" So lon nhat MAX = %5.2f\n",max);
printf(" So nho nhat MIN = %5.2f\n",min);
getch();
return;
}

```

Ta được kết quả chương trình CT49 như sau :

*Chuong trinh tim MAX va MIN trong ma tran*

*Cho hang cua ma tran m = 2*

*Cho cot cua ma tran n = 2*

Vao ma tran A

$A[1,1] = 15$

$A[1,2] = 35$

$A[2,1] = 76$

$A[2,2] = 28$

So lon nhat MAX = 76.00

So nho nhat MIN = 15.00

**Bài số 50** Tính tổng của hai ma trận

**Bài giải** Tổng của hai ma trận C = A + B có các phần tử :

$$c_{ij} = a_{ij} + b_{ij}$$

```
/* CT50 Chuong trinh tinh tong hai matran */
```

```
#include <stdio.h>
```

```
typedef float mt[10][10];
```

```
main()
```

```
{
```

```
int i,j,k,n,m;
```

```
float x;
```

```
mt a,b,c;
```

```
/* Vao ma tran A */
```

```
printf(" Cho cap cua ma tran n = " );scanf("%d",&n);
```

```
printf(" m = " );scanf("%d",&m);
```

```
for (i=1;i<=n;i++)
```

```
    for (j=1;j<=m;j++)
```

```
{
```

```
    printf("\n a[%d][%d] = ",i,j);
```

```
    scanf("%f",&x);
```

```

    a[i][j] = x ;
}
printf("\n");
/* Vao ma tran B */
for (i=1;i<=n;i++)
    for (j=1;j<=m;j++)
    {
        printf("\n b[%d][%d] = ",i,j);
        scanf("%f",&x);
        b[i][j] = x;
    }
printf("\n");
/* Tinh tong a+b */
for (i=1;i<=n;i++)
    for (j=1;j<=m;j++)
    {
        c[i][j] = a[i][j]+b[i][j];
    }
/* Hien thi ma tran A tren man hinh */
printf(" Ma tran A\n\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=m;j++)
        printf("%4.2f ",a[i][j]);
    printf("\n");
    printf("\n");
}

```

```

/* Hien thi ma tran B tren man hinh */
printf(" Ma tran B\n\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=m;j++)
        printf("%4.2f ",b[i][j]);
    printf("\n");
    printf("\n");
}
printf("\n");

/* Hien thi ma tran tong tren man hinh */
printf(" Ma tran ket qua\n\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=m;j++)
        printf("%4.2f ",c[i][j]);
    printf("\n");
    printf("\n");
}
getch();
return;
}

```

Kết quả chương trình CT50 như sau :

*Cho cap cua ma tran n = 2 m = 2*

*a[1][1] = 23*

*a[1][2] = 55*

*a[2][1] = 10*

*a[2][2] = 25.75*

$b[1][1] = -34.25$

$b[1][2] = 37.68$

$b[2][1] = 75.25$

$b[2][2] = 56.25$

*Ma tran A*

$23.00 \ 55.00$

$10.00 \ 25.75$

*Ma tran B*

$-34.25 \ 37.68$

$75.25 \ 56.25$

*Ma tran ket qua*

$-11.25 \ 92.68$

$85.25 \ 82.00$

**Bài số 51** Tính ma trận tích  $C = A.B$

**Bài giải** Các phần tử  $c_{ij}$  của ma trận tích được xác định như sau :

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}.$$

Việc tính  $c_{ij}$  dựa về bài toán tính tổng của một dãy số. Cần phải thực hiện ba chu trình for để biến đổi các chỉ số i,j,k

/\* CT51 Chuong trinh nhan ma tran tong quat \*/

```
#include <stdio.h>
```

```
typedef float mt[10][10];
```

```
main()
```

```
{
```

```
int i,j,k,m,n,p;
```

```
float x;
```

```

mt a,b,c;
/* Vao ma tran A m cot va n hang */
printf(" Vao ma tran A m cot n hang :\n\n");
printf(" Cho so cot cua ma tran m = " );
scanf("%d",&m);
printf(" Cho so hang cua ma tran n = " );
scanf("%d",&n);
for (i=1;i<=m;i++)
    for (j=1;j<=n;j++)
    {
        printf("\n a[%d][%d] = ",i,j);
        scanf("%f",&x);
        a[i][j] = x;
    }
printf("\n");
/* Vao ma tran B n cot p hang */
printf(" Vao ma tran B n cot p hang :\n\n");
printf(" Cho so cot cua ma tran n = " );
scanf("%d",&n);
printf(" Cho so hang cua ma tran p = " );
scanf("%d",&p);
for (i=1;i<=n;i++)
    for (j=1;j<=p;j++)
    {
        printf("\n b[%d][%d] = ",i,j);
        scanf("%f",&x);
        b[i][j] = x;
    }

```

```

printf("\n");
/* Tinh tich a*b */
for (i=1;i<=m;i++)
    for (j=1;j<=p;j++)
    {
        c[i][j] = 0;
        for (k=1;k<=n;k++)
            c[i][j] += a[i][k]*b[k][j];
    }
/* Hien thi ma tran A tren man hinh */
printf(" Ma tran A\n\n");
for (i=1;i<=m;i++)
{
    for (j=1;j<=n;j++)
        printf("%4.2f ",a[i][j]);
    printf("\n");
    printf("\n");
}
/* Hien thi ma tran B tren man hinh */
printf(" Ma tran B\n\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=p;j++)
        printf("%4.2f ",b[i][j]);
    printf("\n");
    printf("\n");
}
printf("\n");

```

```

/* Hien thi ma tran tich tren man hinh */
printf(" Ma tran ket qua\n\n");
for (i=1;i<=m;i++)
{
    for (j=1;j<=p;j++)
        printf("%4.2f ",c[i][j]);
    printf("\n");
    printf("\n");
}
getch();
return;
}

```

Kết quả chương trình CT51 như sau :

Vao ma tran A m cot n hang :

Cho so cot cua ma tran m = 2

Cho so hang cua ma tran n = 3

Vao ma tran B n cot p hang :

Cho so cot cua ma tran n = 3

Cho so hang cua ma tran p = 2

*Ma tran A*

**12.00 34.00 56.00**

**78.00 91.00 45.00**

*Ma tran B*

**67.00 45.00**

**23.25 57.75**

**45.00 91.23**

*Ma tran ket qua*

4114.50 7612.38

9366.75 12870.60

**Bài số 52** Tính ma trận tích  $C = A \cdot B$  số phần tử cố định

**Bài giải** Đây là một trường hợp riêng của chương trình nhân ma trận tổng quát CT51. Áp dụng thuật toán tương tự với số hàng và cột cố định ta được kết quả của chương trình CT52.

```
/* CT52 Chuong trinh nhan ma tran so phan tu la co dinh */
#include <stdio.h>
main()
{
    float a[3][2],b[2][4],c[3][4],x;
    int i,j,k;
    /* Vao ma tran A */
    printf(" Vao ma tran A :\n");
    for (i=1;i<=3;++)
        for (j=1;j<=2;++)
    {
        printf("\n a[%d][%d] = ",i,j);
        scanf("%f",&x);
        a[i][j] = x;
    }
    /* Vao ma tran B */
    printf(" Vao ma tran B :\n");
    for (i=1;i<=2;++)
        for (j=1;j<=4;++)
    {
        printf("\n b[%d][%d] = ",i,j);
        scanf("%f",&x);
        b[i][j] = x;
    }
    for (i=1;i<=3;++)
        for (j=1;j<=4;++)
    {
        c[i][j] = 0;
        for (k=1;k<=2;k++)
            c[i][j] += a[i][k] * b[k][j];
    }
    for (i=1;i<=3;++)
    {
        for (j=1;j<=4;j++)
            printf(" %f",c[i][j]);
        printf("\n");
    }
}
```

```

{
printf("\n b[%d][%d] = ",i,j);
scanf("%f",&x);
b[i][j] = x;
}

/* Tinh tich a*b */
for (i=1;i<=3;i++)
for (j=1;j<=4;j++)
{
c[i][j] = 0;
for (k=1;k<=2;k++)
c[i][j] += a[i][k]*b[k][j];
}

/* Hien thi ma tran a tren man hinh */
printf(" Ma tran a\n");
for (i=1;i<=3;i++)
{
for (j=1;j<=2;j++)
printf("%4.2f ",a[i][j]);
printf("\n");
printf("\n");
}
/* Hien thi ma tran b tren man hinh */
printf(" Ma tran b\n");
for (i=1;i<=2;i++)
{
for (j=1;j<=4;j++)
printf("%4.2f ",b[i][j]),
}

```

```

printf("\n");
printf("\n");
}

printf("\n");
/* Hien thi ma tran tich tren man hinh */
printf(" Ma tran ket qua\n");
for (i=0;i<2;i++)
{
    for (j=1;j<4;j++)
        printf("%4.2f ",c[i][j]);
    printf("\n");
    printf("\n");
}
getch();
return;
}

```

*Ma tran a*

24.00 45.00

67.00 -57.50

34.60 47.45

*Ma tran b*

45.20 34.56 20.45 65.12

25.00 68.12 23.00 8.75

*Ma tran ket qua*

25.00 68.12 23.00

2209.80 3894.84 1525.80

**Bài số 53** Tính tích của hai ma trận với các phần tử là phức.

**Bài giải** Tổ chức một cấu trúc số phức có trường là phần thực reel và phần số ảo imag. Xây dựng hàm MT\_tich để tính tích hai ma trận với các phần tử phức.

```
/* CT53 Chuong trinh tinh toan hai ma tran phuc dung pointer */
#define N 3
#define P 2
#define Q 4
typedef struct
{
    double reel;
    double imag;
}sophuc;

main()
{
void MT_tich(sophuc*,sophuc*,sophuc*,int,int,int);
sophuc a[N][P],b[P][Q],c[N][Q];
int i,j;
/* Khoi tao ma tran A */
for (i=0;i<N;i++)
    for (j=0;j<P;j++)
    {
        a[i][j].reel = i+j
        a[i][j].imag = i+2*j
    }
/* Khoi tao ma tran b */
for (i=0;i<P;i++)
    for (j=0;j<Q;j++)
```

```

{ b[i][j].reel = i+j
b[i][j].imag = i+2*j
}

/* Tinh tich a*b */
MT_tich((sophuc*) &a,(sophuc*) &b,(sophuc*) &c,N,P,Q);

/* Hien thi ma tran A tren man hinh */
printf(" Ma tran A\n");
for (i=0;i<N;i++)
{
    for (j=0;j<P;j++)
        printf("(%.4.1f)+j%.4.1f ",a[i][j].reel,a[i][j].imag);
    printf("\n");
}

/* Hien thi ma tran B tren man hinh */
printf(" Ma tran B\n");
for (i=0;i<N;i++)
{
    for (j=0;j<Q;j++)
        printf("(%.4.1f)+j%.4.1f ",b[i][j].reel,b[i][j].imag);
    printf("\n");
}

printf("\n");

/* Hien thi ma tran tich tren man hinh */
printf(" Ma tran Tich\n");
for (i=0;i<N;i++)

```

```

{
for (j=0;j<Q;j++)
    printf("(%.4f+ j%.4f) ",c[i][j].real,c[i][j].imag);
printf("\n");
}
printf("\n");
scanf() /* Dung man hinh de xem ket qua */
return 0 ;
}

void MT_tich(sophuc* a,sophuc* b,sophuc* c,int n,int p,int q)
{
void Tich();
int i,j,k;
sophuc s,pr;
sophuc *aik,*bkj,*cij;
cij = c;
for (i=0;i<n;i++)
    for (j=0;j<q;j++)
    {
        aik = a + i*p;
        bkj = b + j;
        s.real = 0;
        s.imag = 0;
        for (k=0;k<p;k++)
        {
            Tich(aik,bkj&pr);
            s.real += pr.real;
        }
    }
}

```

```

    s.imag += pr.imag;
    aik++;
    bkj += q;
}
*(cij++) = s;
}
} /* cua MT_tich */

void Tich(x,y,prod)
sophuc *x,*y,*prod;
{
prod->reel = x->reel * y->reel - x->imag * y->imag;
prod->imag = x->reel * y->imag + x->imag * y->reel;
}

```

Ta được kết quả chương trình CT53 như sau :

*Ma tran A*

$$\begin{aligned} &(0.0) + j(0.0) \quad (1.0) + j(2.0) \\ &(1.0) + j(1.0) \quad (2.0) + j(3.0) \\ &(2.0) + j(2.0) \quad (3.0) + j(4.0) \end{aligned}$$

*Ma tran B*

$$\begin{aligned} &(0.0) + j(0.0) \quad (1.0) + j(2.0) \quad (2.0) + j(4.0) \quad (3.0) + j(6.0) \\ &(1.0) + j(1.0) \quad (2.0) + j(3.0) \quad (3.0) + j(5.0) \quad (4.0) + j(7.0) \\ &(-1.0) + j(3.0) \quad (-4.0) + j(7.0) \quad (-7.0) + j(11.0) \quad (-10.0) + j(15.0) \end{aligned}$$

*Ma tran Tich*

$$\begin{aligned} &(-1.0) + j(3.0) \quad (-4.0) + j(7.0) \quad (-7.0) + j(11.0) \quad (-10.0) + j(15.0) \\ &(-1.0) + j(5.0) \quad (-6.0) + j(15.0) \quad (-11.0) + j(25.0) \quad (-16.0) + j(35.0) \\ &(-1.0) + j(7.0) \quad (-8.0) + j(23.0) \quad (-15.0) + j(39.0) \quad (-22.0) + j(55.0) \end{aligned}$$

### Bài số 54 Nhập hai ma trận C = A.B kiểu FILE.

**Bài giải** Các thông tin về ma trận được tổ chức thành từng tệp (file)

Dùng hàm fopen (fa,"w") để mở tệp và ghi các dữ liệu là các phần tử của ma trận  $a_{ij}$  và  $b_{ij}$ .

Dùng hàm fclose (fp) để đóng tệp. Giá trị của hàm fopen được gán cho biến fp cần được khai báo trước theo kiểu con trỏ FILE \* fp

Chương trình CT54 bao gồm việc vào ma trận A,B dạng tệp dữ liệu. Xây dựng hàm vào ma trận mtin, ra ma trận mtout, nhân ma trận nhan.

```
/* CT54 Chuong trinh nhan hai ma tran kieu FILE */
#include <stdio.h>
main()
{
FILE *fp;
float x,a[3][2],b[2][4],c[3][4];
char fa[40]; /* Ten tep dai nhat 40 ky tu */
int i,j;
printf("\nVao ma tran A\n");
mtin(a,3,2);
printf("\nVao ma tran B\n");
mtin(b,2,4);
printf("\nVao ten tep du lieu : ");
scanf("%s",fa);
fp = fopen(fa,"w");
for (i=1;i<=3;i++)
    for (j=1;j<=2;j++)
        fprintf(fp,"%f\n",a[i][j]);
fclose(fp);
```

```

fp = fopen(fa,"a");
for (i=1;i<=3;i++)
    for (j=1;j<=4;j++)
        fprintf(fp,"%f\n",b[i][j]);
fclose(fp);
fp = fopen(fa,"r");
for (i=1;i<=3;i++)
    for (j=1;j<=2;j++)
    {
        fscanf(fp,"%f",&x);
        a[i][j] = x;
    }
for (i=1;i<=2;i++)
    for (j=1;j<=4;j++)
    {
        fscanf(fp,"%f",&x);
        b[i][j] = x;
    }
nhan(a,b,c,3,2,4);
sprintf(stdout,"\n Ma tran A\n\n");
mtout(a,3,2);
printf("\n Ma tran B\n\n");
mtout(b,2,4);
printf(stdout,"\n Ma tran Tich\n\n");
mtout(c,3,4);
getch();
return;
}

```

```

/* Cac ham */

mtin(pa,m,n)
float *pa;
int m,n;
{
float x;
int k=m*n,t,i,j
for(t=0;t<k;t++)
{
    i = t/n;
    j = t - i*n;
    printf("\nPhan tu %d,%d = ",i+1,j+1);
    scanf("%f",&x);
    *(pa+t) = x;
}
return;
}

mtout(pa,m,n)
float *pa;
int m,n;
{
int k=m*n,t,s;
for(t=0;t<k;++)
{
    s = (t+1)%n;
    printf("%4.1f%c",*(pa+t),s?" :\n");
}

```

```

printf("\n\n");
return;
}

nhan(pa,pb,pc,m,n,k)
float *pa,*pb,*pc;
int m,n,k;
{
int i,jt;
for(i=0;i<m;++)
for(j=0;j<k;++)
{
*(pc+i*k+j) =0;
for(t=0;t<n;++)
*(pc+i*k+j) +=(*(pa+i*n+t))*(*(pb+t*k+j));
}
return;
}

```

Kết quả chương trình CT54 như sau :

*Vao ma tran A*  
*Phan tu 1,1 = 1*  
*Phan tu 1,2 = 3*  
*Phan tu 2,1 = 5*  
*Phan tu 2,2 = 6*  
*Phan tu 3,1 = 0*  
*Phan tu 3,2 = 2.5*

*Vao ma tran B*

*Phan tu 1,1 = 2*

*Phan tu 1,2 = 0*

*Phan tu 1,3 = 4*

*Phan tu 1,4 = 2.4*

*Phan tu 2,1 = 3*

*Phan tu 2,2 = 6*

*Phan tu 2,3 = -4*

*Phan tu 2,4 = 5*

*Ma tran A*

1.0 3.0

5.0 6.0

0.0 2.5

*Ma tran B*

2.0 0.0 4.0 2.4

3.0 6.0 -4.0 5.0

*Ma tran tich*

11.0 18.0 -8.0 17.4

28.0 36.0 -4.0 42.0

7.5 15.0 -10.0 12.5

**Bài số 55** Chương trình vào ma trận tổng quát dùng FILE

**Bài giải** Chương trình CT55 xử lý vào hai ma trận A và B con trả  
kiểu FILE dùng fopen để mở file ghi dữ liệu và fclose để đóng file.

```
/* CT55 Chuong trinh vao ma tran tong quat voi file */
```

```
#include <stdio.h>
```

```
typedef float mt[50][50];
```

```

void main()
{
FILE *fs11,*fs12,*fs13,*fopen(),*tg;
mt a,b,c;
float x;
int i,jm,n;
/* Mo file ghi du lieu */
printf("\nVao so hang m, so cot n cua ma tran A\n");
printf("m = ");scanf("%d",&m);
printf("n = ");scanf("%d",&n);
for (i=1;i<=m;i++)
    for (j=1;j<=n;j++)
    {
        printf("\n a[%d][%d] = ",i,j);
        scanf("%f",&x);
        a[i][j] = x;
    }
fs11 = fopen("mata.dat","w");
for (i=1;i<=m;i++)
    for (j=1;j<=n;j++)
        fprintf(fs11,"%f\n",a[i][j]);/* Dac ta \n doc du lieu trong file */
fclose(fs11);
getch();
/* Doc du lieu tu file */
tg = fopen("mata.dat","r");
for (i=1;i<=m;i++)
    for (j=1;j<=n;j++)

```

```

{
fscanf(tg,"%f",&x);
a[i][j] = x;
}

fsl2=fopen("matb.dat","w");
printf("\nVao so hang n, so cot p cua ma tran B\n");
printf("n = ");scanf("%d",&n);
printf("p = ");scanf("%d",&p);
for (i=1;i<=n;i++)
    for (j=1;j<=p;j++)
    {
        printf("\n b[%d][%d] = ",i,j);
        scanf("%f",&x);
        b[i][j] = x;
    }
for (i=1;i<=n;i++)
    for (j=1;j<=p;j++)
        fprintf(fsl1,"%f\n",b[i][j]);/* Dac ta \n doc du lieu trong file */
fclose(fsl2);
getch();
return;
}

```

Kết quả ta được :

Vao so hang m, so cot n cua ma tran A

$m = 2$

$n = 2$

$a[1][1] = 1$

$a[1][2] = 2$

$a[2][1] = 3$

$a[2][2] = 4$

vào số hàng  $n$ , số cột  $p$  của ma trận  $B$

$n = 2$

$p = 2$

$b[1][1] = 0$

$b[1][2] = -1$

$b[2][1] = 4$

$b[2][2] = 2$

**Bài số 56** Tính định thức của một ma trận.

**Bài giải** Để tìm giá trị của định thức  $D$  ta biến đổi ma trận sao cho các phần tử dưới đường chéo chính bằng không, do đó giá trị của định thức bằng tích của các phần tử trên đường chéo này

$$D(A) = (-1)^p a_{11} a_{22} a_{33} \dots a_{nn}$$

với  $p$  là tổng các phép hoán vị đã được thực hiện.

```
/* CT56 Chuong trinh tinh dinh thuc cap n */
```

```
#include <stdio.h>
```

```
#define TRUE 1
```

```
#define FALSE 0
```

```
main()
```

```
{
```

```
int i,j,k,n,ok1,ok2;
```

```
float d,c,x,a[10][10];
```

```

printf(" Tinh dinh thuc cap n\n");
printf(" Vao bac cua dinh thuc n = ");
scanf("%d",&n);
printf(" Vao ma tran A\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=n;j++)
    {
        printf("\n a[%d][%d] = ",i,j);
        scanf("%f",&x);
        a[i][j] = x;
    }
}
/* Hien thi ma tran A tren man hinh */
printf(" Ma tran A\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=n;j++)
        printf("%4.2f ",a[i][j]);
    printf("\n");
}
d=1;
i=1;
ok2=TRUE;
while (ok2 && (i<=n))
{
    if (a[i][i]==0)
    {
        ok1=TRUE;
        k=i+1;

```

```

while (ok1 && (k<=n))
{
    if (a[k][i]!=0)
    {
        for (j=i;j<=n;j++)
        {
            /* Doi dong i cho dong k */
            c=a[i][j];
            a[i][j]=a[k][j];
            a[k][j]=c;
        }
        d = -d;
        ok1=FALSE;
    }
    else k++;
    if (k>n)
    {
        printf(" MA TRAN SUY BIEN\n");
        ok2=FALSE;
        d=0;
    }
}
if (a[i][i]== 0)
{
    c=a[i][i];
    for (j=i+1;j<=n;j++) a[i][j]=a[i][j]/c;
    for (k=i+1;k<=n;k++)
    {

```

```

c=a[k][i];
for (j=i+1;j<=n;j++) a[k][j]=a[k][j]-a[i][j]*c;
}
}
i=i+1;
}
while(ok2 && (i<=n))
{
if (a[i][i]!= 0)
{
c=a[i][i];
for (j=i+1;j<=n;j++) a[i][j]=a[i][j]/c;
for (k=i+1;k<=n;k++)
{
c=a[k][i] ;
for (j=i+1;j<=n;j++) a[k][j]=a[k][j]-a[i][j]*c;
}
i++;
}
}
if (ok2)
{
for (i=1;i<=n;i++)
d*= a[i][i];
printf(" GIA TRI DINH THUC D = %10.2f\n",d);
}
}
getch();

```

```
return;  
}
```

Kết quả của chương trình CT56 như sau :

*Tinh dinh thuc cap n*

*Vao bac cua dinh thuc n = 2*

*Vao ma tran A*

*a[1][1] = 2*

*[1][2] = 4.5*

*a[2][1] = 3.6*

*a[2][2] = -1.5*

*Ma tran A*

*2.00 4.50*

*3.60 -1.50*

*GIA TRI DINH THUC D = -19.20*

### Bài số 57 Tính ma trận nghịch đảo

**Bài giải** Nghịch đảo của ma trận A ký hiệu là  $A^{-1}$  sao cho  $A \cdot A^{-1} = 1$ .

Có nhiều thuật toán khác nhau để tính ma trận nghịch đảo. Một trong các thuật toán là :

Nếu bên cạnh ma trận A có ma trận đơn vị, chia các hàng của ma trận này cho cùng một hệ số rồi trừ đi một số hàng khác để ma trận A trở thành ma trận đơn vị thì kết quả ma trận mới tạo nên là nghịch đảo của ma trận A.

```
/* CT57 Chuong trinh tinh ma tran nghich dao */
```

```
#include <stdio.h>
```

```
typedef float mt[10][10];
```

```

void main()
{
int t,t1,i,j,k,n;
float c,x;
mt a,b;
printf(" TINH MA TRAN NGHICH DAO\n");
printf(" Vao bac cua ma tran n = ");
scanf("%d",&n);
printf("Vao ma tran ban dau a\n");
for (i=1;i<=n;i++)
    for (j=1;j<=n;j++)
    {
        printf("a[%d][%d] = ",i,j);
        scanf("%f",&x);
        a[i][j] = x;
    }
printf("\n");
for (i=1;i<=n;i++)
    for (j=n+1;j<=2*n;j++)
    {
        if (j==i+n) a[i][j] = 1;
        else a[i][j] = 0;
    }
i=1;
t1=1;
while (t1 && (i<=n))
{
    if (a[i][i]==0)

```

```

{
t=1;
k=i+1;
while (t && (k<=n))
{
if (a[k][i]!=0)
{
for (j=1;j<=2*n;j++)
{
c = a[i][j];
a[i][j] = a[k][j];
a[k][j] = c;
}
t=0;
}
else k=k+1;
if (k==n+1)
{
if (a[i][k-1]==0)
{
printf(" Ma tran suy bien \n");
t1=0;
}
}
i++;
}
if (a[i][i]!=0)
{

```

```

c = a[i][i];
for (j=i;j<=2*n;j++)
    a[i][j] = a[i][j]/c;
}

for (k=1;k<=n;k++)
{
    if (k!=i)
    {
        c = a[k][i];
        for (j=i;j<=2*n;j++)
            a[k][j]=a[k][j]-a[i][j]*c;
    }
}

i++;
}

while(t1 && (i<=n))
{
    if (a[i][i]!=0)
    {
        c = a[i][i];
        for (j=i;j<=2*n;j++)
            a[i][j] = a[i][j]/c;
    }

    for (k=1;k<=n;k++)
    {
        if (k!=i)
        {
            c = a[k][i];

```

```

        for (j=i;j<=2*n;j++)
            a[k][j]=a[k][j]-a[i][j]*c;
    }
}
i++;
}
if (t1)
{
    printf(" MA TRAN KET QUA\n");
    for (i=1 ;i<=n ;i++)
    {
        for (j=n+1 ;j<=2*n ;j++)
            printf("%d[%d] = %10.2f ",i,j,a[i][j]);
        printf("\n");
    }
}
getch();
return;
}

```

Kết quả chương trình CT57 như sau :

*TINH MA TRAN NGHICH DAO*

Vao bac cua ma tran n = 2

Vao ma tran ban dau a

$a[1][1] = 2$

$a[1][2] = 1.5$

$a[2][1] = 3.6$

$a[2][2] = -4.3$

### MA TRAN KET QUA

$$\begin{array}{ll} a[1][3] = & 0.31 \quad a[1][4] = & 0.11 \\ a[2][3] = & 0.26 \quad a[2][4] = & -0.14 \end{array}$$

### Bài số 58 Giải hệ phương trình đại số tuyến tính bậc n

**Bài giải** Có nhiều thuật toán khác nhau để giải hệ phương trình đại số bậc n. Phương pháp khử ẩn Gauss như sau :

Dùng phương trình đầu tiên để khử  $x_1$  trong  $n - 1$  phương trình còn lại.

Nếu  $a_{11} \neq 0$  ta tính  $n - 1$  nhân thức  $m_i = \frac{a_{i1}}{a_{11}}$ ,  $i = 2, 3, \dots, n$

Nhân phương trình đầu với  $m_i$  và trừ vào phương trình i với  $i = 2, 3, \dots, n$ . Tiếp theo ta được các hệ số của phương trình :

$$a_{ij}^{(1)} = a_{ij} - m_i a_{1j}, \quad i = 2, 3, \dots, n.$$

$$b_i^{(1)} = b_i - m_i b_1, \quad j = 1, 2, \dots, n.$$

trong đó  $a_{11}^{(1)} = 0$ , ẩn số  $x_1$  đã bị khử trong  $n - 1$  phương trình của hệ thống ban đầu.

Phương trình thứ hai đã biến đổi trong bước đầu tiên được dùng để khử ẩn  $x_2$  trong  $n - 2$  phương trình còn lại.

Ở bước k ẩn số  $x_k$  được khử với nhân thức

$$m_i^{(k-1)} = \frac{a_{ij}^{(k-1)}}{a_{kk}^{(k-1)}}, \quad i = k+1, k+2, \dots, n$$

trong đó  $a_{kk}^{(k-1)} \neq 0$  và  $a_{ij}^{(k)} = a_{ij}^{(k-1)} - m_i^{(k-1)} a_{kj}^{(k-1)}$ , ( $j = k, \dots, n$ )

$$b_i^{(k)} = b_i^{(k-1)} - m_i^{(k-1)} b_k^{(k-1)}, \quad (i = k+1, \dots, n)$$

k lấy giá trị liên tiếp từ 1, 2 đến  $n - 1$ . Trong giai đoạn  $k = n - 1$  ẩn số  $x_{n-1}$  bị khử trong phương trình cuối cùng của hệ phương trình. Cuối cùng ta được một hệ phương trình dạng chéo trong đó phương trình cuối cùng chỉ chứa một ẩn  $x_n$ .

$$\begin{aligned}
 a_{11}x_1 + a_{12}x_2 + \dots + a_{1k}x_k + \dots + a_{1n}x_n &= b_1 \\
 a_{21}^{(1)}x_2 + \dots + a_{2k}^{(1)}x_k + \dots + a_{2n}^{(1)}x_n &= b_2^{(1)} \\
 a_{kk}^{(k-1)}x_k + \dots + a_{kn}^{(k-1)}x_n &= b_k^{(k-1)} \\
 a_{nn}^{(n-1)}x_n &= b_n^{(n-1)}
 \end{aligned}$$

Hệ phương trình có dạng chéo, ta dễ dàng tính :

$$\begin{aligned}
 x_n &= b^{(n-1)}/a_{nn}^{(n-1)} \\
 x_{n-1} &= (b_{n-1}^{(n-2)} - a_{n-1,n}^{(n-2)}x_n)/a_{n-1,n-1}^{(n-2)} \\
 &\dots \\
 x_j &= (b_j^{(j-1)} - a_{j+1,j}^{(j-1)}x_{j+1})/a_{jj}^{(j-1)} \\
 j &= n-2, \dots, 3, 2, 1
 \end{aligned}$$

Chương trình CT58 gồm việc cho bậc phương trình, vào các hệ số  $a_{ij}$ ,  $b_j$  và tìm các điều kiện để hệ phương trình tuyến tính bậc  $n$  có nghiệm, nếu không sẽ thông báo hệ vô định, hoặc vô nghiệm và in nghiệm của hệ phương trình.

```

/* CT58 Chuong trinh giai he PTTT bac n */
#include <stdio.h>
typedef float mt[20][20];
typedef float vec[21];

void main()
{
    int t,tl,i,j,k,n;
    float c,z;
    mt a;
    vec x;
    printf("\n He PTTT bac n ");

```

```

printf("\n Vao bac cua phuong trinh n = ");
scanf("%d",&n);
printf("\n Vao ma tran he so \n ");
for (i=1;i<=n;i++)
    for (j=1;j<=n+1;j++)
    {
        printf("a[%d][%d] = ",i,j);
        scanf("%f",&z);
        a[i][j] = z;
    }
i=1 ;
t1=1 ;
while (t1 && (i<n))
{
    t=1;
    k=i+1;
    while (t && (k<=n))
    {
        if (a[k][i]!=0)
        {
            for (j=1;j<=n+1;j++)
            {
                c = a[i][j];
                a[i][j] = a[k][j];
                a[k][j] = c;
            }
            t=0;
        }
    }
}

```

```

else k=k+1;
if (k==n+1)
{
    if (a[i][k]==0)
        printf("\n He vo dinh ");
    else printf("\n He vo nghiem ");
    t1=0;
}
if (a[i][i]!=-0)
{
    c = a[i][i];
    for (j=i+1;j<=n+1;j++)
        a[i][j] = a[i][j]/c;
    for (k=i+1;k<=n;k++)
    {
        c = a[k][i];
        for (j=i+1;j<=n+1;j++)
            a[k][j]=a[k][j]-a[i][j]*c;
    }
    i = i+1;
}
if (t1)
{
    x[n] = a[n][n+1]/a[n][n];
    for (i=n-1;i>=1;i--)
    {

```

```

x[i] = a[i][n+1];
for(j=n;j>=i+1;j--)
    x[i] = x[i]-a[i][j]*x[j];
}
printf("\n Ma tran ket qua \n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=n+1;j++)
        printf("a[%d][%d] = %10.3f ",i,j,a[i][j]);
    printf("\n");
}
printf("\n Nghiem cua he PTTT bac n ");
for(i=1;i<=n;i++)
    printf("\n x[%d] = %10.3f",i,x[i]);
}
getch(); —
return;
}

```

Ta được kết quả của phương trình CT58 như sau :

$a[1][1] = -2.5$   
 $a[1][2] = 4.1$   
 $a[1][3] = 6.2$   
 $a[1][4] = 5.25$   
 $a[2][1] = 1.75$   
 $a[2][2] = 3.8$   
 $a[2][3] = 4$

$$a[2][4] = 1.25$$

$$a[3][1] = 12$$

$$a[3][2] = 3.4$$

$$a[3][3] = 1.7$$

$$a[3][4] = 2.5$$

*Ma tran ket qua*

$$a[1][1] = 1.750 \quad a[1][2] = 2.171 \quad a[1][3] = 2.286 \quad a[1][4] = \\ 0.714$$

$$a[2][1] = 12.000 \quad a[2][2] = -22.657 \quad a[2][3] = 1.136 \quad a[2][4] = \\ 0.268$$

$$a[3][1] = -2.500 \quad a[3][2] = 9.529 \quad a[3][3] = 1.094 \quad a[3][4] = \\ 4.482$$

*Nghiem cua he PTTT bac n*

$$x[1] = 0.870$$

$$x[2] = -4.385$$

$$x[3] = 4.097$$

## Chương 5

# ĐA THỨC VÀ NỘI SUY ĐA THỨC

**Bài số 59** Tính giá trị của một đa thức tại điểm  $x_0$  theo thuật toán Horner.

**Bài giải** Cho đa thức  $P(x) = a_0x^n + a_1x^{n-1} + \dots + a_{n-1}x + a_n$ . Cho bậc của đa thức, cần đưa các dữ liệu vào là các hệ số  $a_0, a_1, \dots, a_n$  và  $x_0$ .

Tính giá trị của đa thức theo thuật toán Horner

$$P = (((a_0x + a_1)x + a_2)x + \dots + a_{n-1})x + a_n$$

với  $P_k = P_{k-1}x + a_k ; k = 1, 2, \dots, n$ ;  $P_0 = a_0$ . Xác định hàm DaThuc để tính giá trị của đa thức theo thuật toán Horner.

```
/* CT59 Chuong trinh tinh gia tri da thuc theo Horner */
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
#define MAXHESO 21
typedef float heso[MAXHESO]; /* So cac he so cua da thuc */
int n;
heso a;
float xo;
```

```

float DaThuc(heso, float ,int );

void main()
{
    int i;
    printf("\n** TINH GIA TRI DA THUC THEO HORNER **\n");
    printf("\nCho mot gia tri Xo : ");
    scanf("%f",&x0);
    printf("\nCho bac cua da thuc n = ");
    scanf("%d",&n);
    for (i=0;i<=n;i++)
    {
        printf("a[%d] = ",i);
        scanf("%f",&a[i]);
    }
    printf("\nGIA TRI DA THUC = %6.3f\n",DaThuc(a,x0,n));
    getch();
}

float DaThuc(heso b,float XXo,int n)
{
    int i;
    float tg;
    tg = b[n];
    for (i=n;i>1;--i)
        tg = b[i-1]+XXo*tg;
    return(tg);
}

```

Ta được kết quả của chương trình CT59 như sau :

**\*\* TINH GIA TRI DA THUC THEO HORNER \*\***

Cho mot gia tri Xo : 1

Cho bac cua da thuc n = 3

a[0] = 2

a[1] = 1.5

a[2] = 4.25

a[3] = 2.75

GIA TRI DA THUC = 8.500

**Bài số 60** Tính tổng của hai đa thức.

**Bài giải** Cho hai đa thức :

$$A(x) = a_1x^n + a_2x^{n-1} + \dots + a_nx + a_{n+1} \text{ bậc } n \text{ và}$$

$$B(x) = b_1x^m + b_2x^{m-1} + \dots + b_mx + b_{m+1} \text{ bậc } m \quad (m \leq n)$$

Tổng C(x) = A(x) + B(x) là một đa thức bậc lớn nhất là n.

Các hệ số  $c_k$  của đơn thức  $n-k+1$  của C(x) là tổng của các đơn thức  $n-k+1$  của A(x) và B(x).

Tổng quát ta được các hệ số  $c_k$ ,  $k = 1, \dots, n+1$  của đa thức C(x) :

$$\left\{ \begin{array}{l} i = n-m \\ \left[ \begin{array}{l} \text{nếu } k \leq i, c_k = a_k \\ \text{nếu không } c_k = a_k + b_k \end{array} \right] \quad | \quad k = 1, \dots, n+1 \end{array} \right.$$

*/\* CT60 Chuong trinh cong da thuc \*/*

```
#include <stdio.h>
```

```
void main ()
```

```
{
```

```
float a[20],b[20],c[20];
```

```
int i,k,n,np,m,mp;
```

```

printf("\nCong hai da thuc A(x) bac n va B(x) bac m\n");
printf("Cho bac cua da thuc A(x) n = ");
scanf("%d",&n);
printf("Cho bac cua da thuc B(x) m = ");
scanf("%d",&m);
np = n+1;
mp = m+1;
printf("\n Vao cac he so cua da thuc A(x)\n");
for (i=1;i<=np;i++)
{
    printf("A(%d) = ",i);
    scanf("%f",&a[i]);
}
printf("\n Vao cac he so cua da thuc B(x)\n");
for (i=1;i<=mp;i++)
{
    printf("B(%d) = ",i);
    scanf("%f",&b[i]);
}
i = np - mp;
for (k=1;k<=np;k++)
{
    if (k<=i) c[k] = a[k];
    else c[k] = a[k] + b[k-1];
}
printf(" Da thuc A(x) la \n");
for (i=1;i<=np;i++)
{
    printf("A(%d) = %4.2f,%i,a[i]);
    printf("\n");
}
printf(" Da thuc B(x) la \n");

```

```

for (i=1;i<=mp;i++)
    printf("B(%d) = %4.2f",i,b[i]);
printf("\n");
printf(" Da thuc tong C(x) la \n");
for (k=1;k<=np;k++)
    printf("c(%d) = %4.2f",k,c[k]);
printf("\n");
getch();
return;
}

```

Kết quả chương trình CT60 như sau :

*Cong hai da thuc A(x) bac n va B(x) bac m*

*Cho bac cua da thuc A(x) n = 2*

*Cho bac cua da thuc B(x) m = 2*

*Vao cac he so cua da thuc A(x)*

*A(1) = 1*

*A(2) = 2*

*A(3) = 3*

*Vao cac he so da thuc B(x)*

*B(1) = 4*

*B(2) = 5*

*B(3) = 6*

*Da thuc A(x) la*

*A(1) = 1.00, A(2) = 2.00, A(3) = 3.00,*

*Đa thức B(x) là*

$$B(1) = 4.00, B(2) = 5.00, B(3) = 6.00,$$

*Đa thức tong C(x) là*

$$c(1) = 1.00, c(2) = 6.00, c(3) = 8.00,$$

**Bài số 61** Tính tích của hai đa thức.

**Bài giải** Tích  $C(x) = A(x).B(x)$  là một đa thức bậc  $l = n + m$ .

Hệ số  $c_k$  của đơn thức bậc  $l - k + 1$  của  $C(x)$  là tổng của tất cả tích các hệ số của các đơn thức bậc  $k_1$  ( $0 \leq k_1 \leq n$ ) của  $A(x)$  và  $k_2$  ( $0 \leq k_2 \leq m$ ) của  $B(x)$ .

Tổng quát ta xác định bậc  $l$  và các hệ số  $c_k$ ,  $k = 1, \dots, l+1$  của  $C(x)$

$$\left\{ \begin{array}{l} l = n + m \\ \left[ \begin{array}{l} \text{nếu } k \leq n + 1, j = k \\ \text{nếu không } j = n + 1 \\ \text{nếu } k \leq m + 1, i_0 = 1 \\ \text{nếu không } i_0 = k - m \\ \quad \quad \quad \sum_{i=i_0}^j a_i b_{k-i+1} \end{array} \right] \quad \quad \quad k = 1, \dots, l + 1 \end{array} \right.$$

*/\* CT61 Chuong trinh nhan hai da thuc \*/*

```
#include <stdio.h>
#define MAXSIZE 50
typedef float vecto[MAXSIZE+1];
void main()
{
    int n,np,m,mp;
    vecto a,b,c;
    int i,io,j,k,lp;
```

```

printf("\nNhan hai da thuc A(x) bac n va B(x) bac m\n");
printf("\nCho bac cua da thuc A(x) n = ");
scanf("%d",&n);
printf("\nCho bac cua da thuc B(x) m = ");
scanf("%d",&m);
np = n+1;
mp = m+1;
printf("\n Vao cac he so cua da thuc A(x)\n");
for (i=1;i<=np;i++)
{
    printf("A(%d) = ",i);
    scanf("%f",&a[i]);
}
printf("\n Vao cac he so cua da thuc B(x)\n");
for (i=1;i<=mp;i++)
{
    printf("B(%d) = ",i);
    scanf("%f",&b[i]);
}
lp = np + mp - 1;
for (k=1;k<=lp;k++)
{
    if (k<=np) j = k;
    else j = np;
    if (k<=mp) io = 1;
    else io = k - mp + 1;
    c[k] = 0.0;
    for (i=io;i<=j;i++)

```

```

c[k] = c[k]+a[i]*b[k-i+1];
}
printf("\nDay he so tich C[k]\n");
for (k=1;k<=lp;k++)
printf("C[%d] = %6.2f\n",k,c[k]);
getch();
return;
}

```

Ta được kết quả chương trình CT61 :

Nhan hai da thuc A(x) bac n va B(x) bac m

Cho bac cua hai da thuc A(x) n = 2

Cho bac cua hai da thuc B(x) m = 2

Vao cac he so cua da thuc A(x)

A(1) = 1

A(2) = 2

A(3) = 3

Vao cac he so cua da thuc B(x)

B(1) = 4

B(2) = 5

B(3) = 6

Day he so tich C[k]

C[1] = 4.00

C[2] = 13.00

C[3] = 28.00

C[4] = 27.00

C[5] = 18.00

### Bài số 62 Chia hai đa thức.

**Bài giải** Cho hai đa thức  $A(x)$  và  $B(x)$ , đa thức tỷ số  $Q(x)$  và phần dư  $R(x)$  của phép chia  $A(x)$  cho  $B(x)$ .

Tổng quát ta xác định bậc l và các hệ số  $q_k$  của đa thức  $Q(x)$  như sau :

$$\left\{ \begin{array}{l} l = n - m \\ r_i^{(0)} = a_i \quad i = 1, \dots, n \\ j_0 = n \\ q_k = \frac{r^{k-1}}{b_1} \\ \left[ \begin{array}{l} \text{nếu } i \leq m, r_i^{(k)} = r_{i+1}^{(k-1)} - q_k b_{i+1} \mid i=1, \dots, j_{k-1} \mid k=1, \dots, l+1 \\ \text{nếu không } r_i^{(k)} = r_{i+1}^{(k-1)} \\ j = j_{k-1} - 1 \end{array} \right. \end{array} \right.$$

*/\* CT62 Chuong trinh chia hai da thuc \*/*

```
#include <stdio.h>
#define MAXSIZE 50
typedef float vecto[MAXSIZE + 1];
void main()
{
    int n,np,m,mp;
    vecto a,b,q,r;
    int i,k;
    int l,lp;
    int j,p;
    float epsi;
    printf("\nChia hai da thuc A(x) bac n va B(x) bac m\n");
    printf("\nCho bac cua da thuc A(x) n = ");
```

```

scanf("%d",&n);
printf("\nCho bac cua da thuc B(x) m = ");
scanf("%d",&m);
np = n+1;
mp = m+1;
printf("\n Vao cac he so cua da thuc A(x)\n");
for (i=1;i<=np;i++)
{
    printf("A(%d) = ",i);
    scanf("%f",&a[i]);
}
printf("\n Vao cac he so cua da thuc B(x)\n");
for (i=1;i<=mp;i++)
{
    printf("B(%d) = ",i);
    scanf("%f",&b[i]);
}
printf("\nCho gia tri epsilon : epsi = ");
scanf("%f",&epsi);
if (mp>1)
{
    lp = np - mp + 1;
    for (i=1;i<=np;i++)
        r[i] = a[i];
    j = np - 1;
    for (k=1;k<=lp;k++)
    {
        q[k] = r[1]/b[1];
        for (l=2;l<=j;l++)
            r[l] = r[l] - q[k]*b[l];
    }
}

```

```

for (i=1;i<=ji++)
{
    if (i<mp)
        r[i] = r[i+1]-q[k]*b[i+1];
    else r[i] = r[i+1];
}
j = j - 1;
}

while ((fabs((double)r[1])<(double)(epsi)) && (j>0))
{
    for (i=1;i<=ji++)
        r[i] = r[i+1];
    j = j-1;
}
if (fabs((double)r[1])<(double)epsi)
    r[1] = 0.0;
jp = j+1;
}

else
{
lp = np;
for (k=1;k<=lp;k++)
    q[k] = a[k]/b[1];
jp = 1;
r[1] = 0.0;
}

printf("\nIn Ket qua He so Q[k]\n");

```

```

for (k=1;k<=lp;k++)
    printf("Q[%d] = %6.2f\n",k,q[k]);
printf("\nIn Ket qua He so R[i]\n");
for (i=1;i<=p;i++)
    printf("R[%d] = %6.2f\n",i,r[i]);
getch();
return;
}

```

Kết quả chương trình CT62 như sau :

*Chia hai da thuc A(x) bac n va B(x) bac m*

*Cho bac cua da thuc A(x) n = 2*

*Cho bac cua da thuc B(x) m = 1*

*Vao cac he so cua da thuc A(x)*

*A(1) = 1*

*A(2) = 2*

*A(3) = 3*

*Vao cac he so cua da thuc B(x)*

*B(1) = 4*

*B(2) = 5*

*Cho gia tri epsilon : epsi = 0.01*

*In ket qua He so Q[k]*

*Q[1] = 0.25*

*Q[2] = 0.19*

*In Ket qua He so R[i]*

*R[1] = 2.06*

**Bài số 63** Tìm giá trị của hàm  $f(x)$  tại điểm  $x_0$  bằng đa thức nội suy Lagrange

**Bài giải** Cho hàm  $f(x)$  xác định bởi cặp các giá trị  $(x_i, y_i)$  với  $i = 1, 2, \dots, n + 1$ . Đa thức nội suy của hàm  $f(x)$  có dạng :

$$P(x) = p_1x^n + p_2x^{n-1} + \dots + p_nx + p_{n+1}$$

sao cho  $P(x_i) = y_i$ ,  $i = 1, 2, \dots, n + 1$ . Cặp  $(x_i, y_i)$  gọi là các điểm nội suy. Ta cần phải xác định các hệ số  $p_1, p_2, \dots, p_{n+1}$  của đa thức nội suy. Về nguyên tắc giải hệ phương trình tuyến tính có thể xác định các hệ số  $p_1, \dots, p_{n+1}$ . Tuy nhiên trong thực tế ta thường sử dụng nội suy đa thức Lagrange xác định như sau :

$$G_k(x_i) = \begin{cases} 1 & \text{nếu } i=k \\ 0 & \text{nếu } i \neq k \end{cases} \quad i = 1, \dots, n+1$$

$$P(x) = y_1G_1(x) + \dots + y_{n+1}G_{n+1}(x)$$

Công thức tính  $G_k(x)$  với mọi  $x$  là :

$$\begin{aligned} G_k(x) &= \frac{(x - x_1) \dots (x - x_{k-1}) (x - x_{k+1}) \dots (x - x_{n+1})}{(x_k - x_1) \dots (x_k - x_{k-1}) (x_k - x_{k+1}) \dots (x_k - x_{n+1})} \\ &= \prod_{\substack{i=1 \\ i \neq k}}^{n+1} \frac{x - x_i}{x_k - x_i}, \quad k = 1, \dots, n+1 \end{aligned}$$

Chương trình CT63 gồm các bước sau :

- Vào dữ liệu các điểm nội suy  $(x_i, y_i)$
- Tính lặp theo công thức Lagrange
- In kết quả

/\* CT63 Chương trình nội suy đa thức Lagrange \*/

```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
```

```

#define NMAX 21
#define TRUE 1
#define FALSE 0
typedef float vecto[NMAX];
int n,i,k,maxKQ;
float po,xo;
vecto x,xx,y,yy,a;
char OK;
void VaoSL(vecto,vecto);
float LapLa(int,vecto,vecto,float);
void InKQ(vecto,vecto);

void main()
{
clrscr();
printf("\n** NOI SUY DA THUC LAGRANGE **");
VaoSL(x,y);
k = 0;
OK = 'C';
do
{
printf("\nTinh gia tri cua Y ung voi Xo = ");
scanf("%f",&xo);
po = LapLa(n,x,y,xo);
printf("\nGia tri cua Y = %6.2f",po);
getch();
k += 1;
maxKQ = k;
}

```

```

xx[k] = xo;
yy[k] = po;
printf("\nCo tinh voi Xo khac nua khong (c/k) : ");
getchar();
if (toupper(getchar())=='C')
    OK = 'C';
else OK = 'K';
}

while (OK=='C');
InKQ(xx,yy) ;
}

void VaoSL(vecto x,vecto y)
{
int i,t;
char *OK;
printf("\nCho ham y = f(x)\n");
printf("\nSo cap (x,y) nhieu nhat la NMAX = 21");
printf("\nCho so diem lap n = ");
scanf("%d",&n);
for (i=1;i<=n;i++)
{
    printf("\nX[%d] = ",i);
    scanf("%f",&x[i]);
    printf("\nY[%d] = ",i);
    scanf("%f",&y[i]);
}
printf("\nBang so lieu cac cap x va y \n") ;

```

```

printf("\n\tNo\t\tX[i]\t\tY[i]");
for (i=1;i<=n;i++)
    printf("\n\t%d\t\t%.1f\t\t%.1f",i,x[i],y[i]);
getch();
/*OK = " ";
t = TRUE;
do
{
    printf("\nCo sua so lieu khong (c/k) : ");
    getchar();
    if (toupper(getchar())=='C')
    {
        printf("\nChi so phan tu can sua i = ");
        scanf("%d",&i);
        printf("\nGia tri moi : X[%d] = ",i);
        scanf("%f",&x[i]);
        printf("\nGia tri moi : Y[%d] = ",i);
        scanf("%f",&y[i]);
    }
    if (toupper(getchar()) !='C') t = FALSE;
}
while (t);
}

float LapLa(int n,vecto x,vecto y,float xo)
{
int i,k;
float go;

```

```

po = 0.0;
for (k=1;k<=n;k++)
{
    go = 1;
    for (i=1 ;i<=n ;i++)
        if (i !=k) go *= ((xo-x[i])/(x[k]-x[i]));
    po += (y[k]*go);
}
return(po);
}

```

```

void InKQ(vecto xx,vecto yy)
{
int i,k ;
printf("\nBang so lieu\n");
printf("\n\tNo\tX[i]\tY[i]");
for (i=1;i<=n;i++)
printf("\n\t%d\t%4.1f\t%4.1f",i,x[i],y[i]);
getch();
printf("\nKet qua tinh toan\n");
for (k=1;k<=maxKQ;k++)
printf("\n\tX = %6.2f\tY = %6.2f",xx[k],yy[k]);
getch();
return;
}

```

Kết quả chương trình CT63 như sau :

Bang so lieu cac cap x va y

No	X[i]	Y[i]
1	1.0	2.0
2	3.0	4.0
3	5.0	6.0
4	7.0	8.0

Co sua so lieu khong ( c/k ) :

k

Tinh gia tri cua Y ung voi Xo = 0

Gia tri cua Y = 1.00

Co tinh voi Xo khac nua khong ( c/k ) : k

**Bài số 64** Tính giá trị gần đúng của hàm  $f(x)$  tại điểm  $x_0$  bằng đa thức nội suy Newton.

**Bài giải** Đa thức nội suy Newton được định nghĩa như sau :

$$P(x) = \delta[x_i] + \sum_{k=1}^n \delta[x_i, \dots, x_{i+k}] \Phi_k(x)$$

Trong đó  $\delta[x_i, \dots, x_{i+k}] = a_i^{(k)}$  với  $1 \leq i \leq n+1-k$  và  $0 \leq k \leq n$ ;

$$\delta[x_i, \dots, x_{n+2-j}] = a_j \text{ với } 1 \leq j \leq n+1$$

là các hệ số của đa thức Newton và được tính như sau :

$$a_i^{(0)} = y_i, i = 1, \dots, n+1$$

$$a_i^{(k)} = \frac{a_{i+1}^{(k-1)} - a_i^{(k-1)}}{x_{i+k} - x_i} \quad \left| \begin{array}{l} i = 1, \dots, n+1-k \\ k = 1, \dots, n \end{array} \right.$$

$$a_{n+2-k} = a_i^{(k-i)}$$

$$a_j = a_i^{(n)}$$

Viết lại đa thức nội suy Newton theo các hệ số  $a_i$  :

$$P(x) = a_1\Phi_n(x) + a_2\Phi_{n-1}(x) + \dots + a_n\Phi_1(x) + a_{n+1}$$

với :  $\Phi_i(x) = (x-x_1)(x-x_2) \dots (x-x_i)$ ,  $i=1,\dots,n+1$

Chương trình CT64 gồm các bước sau đây :

- Vào dữ liệu là cặp các điểm nội suy ( $x_i, y_i$ )
- Dùng hàm tính các hệ số  $a_i$
- Tính lắp theo công thức Newton và in kết quả

```
/* CT64 Chuong trinh noi suy da thuc Newton */
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
#define NPMAX 21
#define TRUE 1
#define FALSE 0
typedef float vecto[NPMAX];
int n,maxKQ;
vecto x,y,a,xx,yy;
float xo;
void VaoSL(vecto,vecto);
void HeSoNEW(int,vecto,vecto,vecto);
float LAPNEW(int,vecto,vecto,float);
void InKQ(vecto,vecto);

void main()
{
    int k;
    float po;
    char OK;
```

```

clrscr();
printf("\n** NOI SUY DA THUC NEWTON **");
VaoSL(x,y);
k = 0;
OK = 'C';
do
{
    printf("\nTinh gia tri cua Y ung voi Xo = ");
    scanf("%f",&x0);
    po = LAPNEW(n,x,y,x0);
    printf("\nGia tri cua Y = %6.2f",po);
    getch();
    k += 1;
    maxKQ = k;
    xx[k] = x0 ;
    yy[k] = po ;
    printf("\nCo tinh voi Xo khac nua khong (c/k) : ");
    getchar();
    if (toupper(getchar())=='C') OK = 'C';
    else OK = 'K';
}
while (OK=='C');
InKQ(xx,yy);
}

void VaoSL(vecto x,vecto y)
{
int i,t;

```

```

printf("\nCho ham y = f(x)\n");
printf("\nSo cap (x,y) nhieu nhat la NMAX = 21");
printf("\nCho so diem lap n = ");
scanf("%d",&n);
for (i=1;i<=n;i++)
{
    printf("\nX[%d] = ",i);
    scanf("%f",&x[i]);
    printf("\nY[%d] = ",i);
    scanf("%f",&y[i]);
}
printf("\nBang so lieu cac cap x va y \n");
printf("\n\tNo\tX[i]\tY[i]");
for (i=1;i<=n;i++)
printf("\n\t%d\t%4.1f\t%4.1f",i,x[i],y[i]);
getch();
t = TRUE;
do
{
    printf("\nCo sua so lieu khong (c/k) : ");
    getchar();
    if (toupper(getchar())=='C')
    {
        printf("\nChi so phan tu can sua i = ");
        scanf("%d",&i);          được
        printf("\nGia tri moi : X[%d] = ",i);
        scanf("%f",&x[i]);
        printf("\nGia tri moi : Y[%d] = ",i);
    }
}

```

```

    scanf("%f",&y[i]);
}
if (toupper(getchar()) !='C') t = FALSE;
}

while (t);
}

void HeSoNEW(int n,vecto x,vecto y,vecto a)
{
int i,k;
float a1;
a = y;
for (k=1;k<=n-1;k++)
{
    a1 = a[1];
    for (i=1;i<=n-k;i++)
        a[i] = (a[i+1]-a[i])/(x[i+k]-x[i]);
    a[n-k+1] = a1;
}
}

float LAPNEW(int n,vecto a,vecto x,float Xo)
{
int k;
float Bo;
Bo = a[1];
for (k=2;k<=n;k++)
    Bo = Bo*(Xo-x[n-k+1])+a[k];
}

```

```

return(Bo);
}

void InKQ(vecto xx,vecto yy)
{
int i,k;
printf("\nBang so lieu\n");
printf("\n\tNo\tX[i]\tY[i]");
for (i=1;i<=n;i++)
    printf("\n\t%d\t%4.1f\t%4.1f",i,x[i],y[i]);
getch();
printf("\nKet qua tinh toan\n");
for (k=1;k<=maxKQ;k++)
    printf("\n\tX = %6.2f\tY = %6.2f",xx[k],yy[k]);
getch();
}

```

Ta được kết quả chương trình CT64 như sau :

**\*\* NOI SUY DA THUC NEWTON \*\***

*Cho ham y = f(x)*

*So cap ( x,y ) nhieu nhat la NMAX = 21*

*Cho so diem lap n = 3*

*X[1] = 1*

*Y[1] = 2*

*X[2] = 3*

*Y[2] = 4*

*X[3] = 6*

*Y[3] = 7*

Bang so lieu cac cap x va y

No	X[i]	Y[i]
1	1.0	2.0
2	3.0	4.0
3	6.0	7.0

Tinh gia tri cua Y ung voi X0 = 1.5

Gia tri cua Y = 5.75

**Bài số 65** Tìm giá trị đa thức nội suy bằng ma trận Lagrange.

**Bài giải** Da thuc Lagrange co the viet duoi dang ma tran :

$$G_k(x) = (x^n, x^{n-1}, \dots, x, 1) \begin{pmatrix} C_{1k} \\ C_{2k} \\ \vdots \\ C_{n+1k} \end{pmatrix}$$

Ma tran vuong  $C_{ij}$  bai n + 1 duoc xac dinh nhu sau :

$$C = \begin{pmatrix} C_{11} & C_{12} & \dots & C_{1n+1} \\ C_{21} & C_{22} & \dots & C_{2n+1} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n+11} & C_{n+12} & \dots & C_{n+1n+1} \end{pmatrix}$$

duoc goi la ma tran Lagrange duoc xay dung bai cac gia tri  $x_1$  den  $x_{n+1}$  theo cong thức truy hoi. Da thuc noi suy duoc viet duoi dang ma tran :

$$P(x) = (x^n, x^{n-1}, \dots, x, 1) . C \cdot \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n+1} \end{pmatrix}$$

$$P(x) = (x^n, x^{n-1}, \dots, x, 1) \begin{pmatrix} p_1 \\ p_2 \\ \dots \\ p_{n+1} \end{pmatrix}$$

Ma trận hệ số  $p_i$  được tính như sau :

$$p_i = \sum_{j=1}^{n+1} C_{ij} y_j \quad i = 1, \dots, n+1$$

Chương trình CT65 gồm các bước :

- Vào dãy các điểm nội suy  $(x_i, y_i)$
- Xác định ma trận Lagrange C
- Xác định p
- Tính  $P(x)$  và ghi kết quả.

*/\* CT65 Chương trình nội suy ma trận Lagrange \*/*

```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
#define NMAX 11
#define NPMAX 12
#define TRUE 1
#define FALSE 0
typedef float vecto[NMAX];
typedef float mt[NMAX][NMAX];
int n;
vecto x,y,p;
void VaoSL(vecto,vecto);
```

```

void InMT(int,int,mt);
void MaTranLaX(int,vecto,mt );
void HeSoP(int ,vecto ,vecto ,vecto );
void InKQ(vecto ,vecto ,vecto );

void main()
{
    clrscr();
    printf("\n** DA THUC NOI SUY CUA HAM MA TRAN
LAGRANGE **");
    VaoSL(x,y);
    HeSoP(n,x,y,p);
    getch();
    printf("\nDa thuc noi suy cua HAM F\n");
    InKQ(x,y,p);
    getch();
}

void VaoSL(vecto x,vecto y)
{
    int i,t;
    printf("\nCho ham y = f(x)\n");
    printf("\nSo cap (x,y) nhieu nhat la NMAX = 11");
    printf("\nCho so diem lap n = ");
    scanf("%d",&n);
    for (i=1;i<=n;i++)
    {
        printf("\nX[%d] = ",i);
        scanf("%f",&x[i]);
    }
}

```

```

printf("\nY[%d] = ",i);
scanf("%f",&y[i]);
}

printf("\nBang so lieu cac cap x va y \n");
printf("\n\tNo\tX[i]\tY[i]");
for (i=1;i<=n;i++)
{
    printf("\n\t%d\t%4.1f\t%4.1f",i,x[i],y[i]);
    getch();
    t = TRUE;
    do
    {
        printf("\nCo sua so lieu khong (c/k) : ");
        getchar();
        if (toupper(getchar())=='C')
        {
            printf("\nChi so phan tu can sua i = ");
            scanf("%d",&i);
            printf("\nGia tri moi : X[%d] = ",i);
            scanf("%f",&x[i]);
            printf("\nGia tri moi : Y[%d] = ",i);
            scanf("%f",&y[i]);
        }
        if (toupper(getchar()) !='C') t = FALSE;
    }
    while (t) ;
}

```

```

void InMT(int m,int n,mt heso)
{
int i,j;
printf("\nMA TRAN HE SO LAGRANGE\n");
for (i=1;i<m;i++)
{
    for (j=1;j<n;j++)
        printf("HS[%d][%d] = %4.1f ",i,j,heso[i][j]);
    printf("\n"); /*Xuong dong sau khi het mot hang */
}
}

void MaTranLaX(int n,vecto x,mt heso)
{
int i,j,k;
float lo,qo,uo,vo;
float phi[NPMAX];
phi[1] = 1.0;
phi[2] = -x[1];
for (i=2;i<=n;i++)
{
    uo = phi[1];
    for (j=2;j<=i;j++)
    {
        vo = phi[j];
        phi[j] = vo-uo*x[i];
        uo = vo;
    }
}

```

```

phi[i+1] = -uo*x[i];
}

for (k=1;k<=n;k++)
{
    lo = n;
    for (j=2;j<=n;j++)
        lo = lo*x[k]+(n-j+1)*phi[j];
    qo = phi[1];
    heso[1][k] = qo/lo;
    for (j=2;j<=n;j++)
    {
        qo = qo*x[k]+phi[j];
        heso[j][k] = qo/lo;
    }
}
InMT(jk,heso);
}

```

```

void HeSoP(int n,vecto x,vecto y,vecto p)
{
int i,j;
float s;
mt heso;
MaTranLaX(n,x,heso);
for (i=1;i<=n;i++)
{
    s = 0.0;
    for (j=1;j<=n;j++)

```

```

    s = s + heso[i][j]*y[j];
    p[i] = s;
}
}

void InKQ(vecto x,vecto y,vecto p)
{
int i,k;
printf("\nBang so lieu\n");
printf("\n\tNo\t\tX[i]\t\tY[i]");
for (i=1;i<=n;i++)
    printf("\n\t%d\t\t%4.1f\t\t%4.1f",i,x[i],y[i]);
getch();
printf("\nKet qua tinh toan\n");
for (k=1;k<=n;k++)
    printf("\nHe so cua don thuc bac %d : %6.2f",n-k,p[k]);
getch();
}

```

Ta được kết quả chương trình CT65 :

**\*\* DA THUC NOI SUY CUA HAM MA TRAN LAGRANGE \*\***

Cho ham  $y = f(x)$

So cap (x,y) nhieu nhat la NMAX = 11

Cho so diem lap n = 3

$X[1] = 1$

$Y[1] = 2$

$X[2] = 3$

$$Y[2] = 4$$

$$X[3] = 5$$

$$Y[3] = 6$$

Bang so lieu cac cap x va y

No	X[i]	Y[i]
1	1.0	2.0
2	3.0	4.0
3	5.0	6.0

### MA TRAN HE SO LAGRANGE

$$HS[1][1] = 0.1 \quad HS[1][2] = -0.2 \quad HS[1][3] = 0.1$$

$$HS[2][1] = -0.1 \quad HS[2][2] = 1.5 \quad HS[2][3] = -0.5$$

$$HS[3][1] = 1.9 \quad HS[3][2] = -1.2 \quad HS[3][3] = 0.4$$

Da thuc noi suy cua HAM F

Ket qua tinh toan

He so cua don thuc bac 2 : 0.00

He so cua don thuc bac 1 : 1.00

He so cua don thuc bac 0 : 1.00

**Bài số 66** Tính đạo hàm cấp l của đa thức P(x).

**Bài giải** Cho đa thức bậc n

$$P(x) = p_1x^n + p_2x^{n-1} + \dots + p_{n+1}$$

Đạo hàm cấp l ( $1 \leq l \leq n$ ) của P(x) được xác định như sau :

$$P'(x) = n p_1x^{n-1} + (n-1)p_2x^{n-2} + \dots + p_n$$

$$P''(x) = n(n-1) p_1x^{n-2} + (n-1)(n-2)p_2x^{n-3} + \dots + 2p_{n-1}$$

$$\text{Tổng quát : } P^{(l)}(x) = d_1x^m + d_2x^{m-1} + \dots + d_{m+1}$$

Trong đó  $m = n - l$ ,  $d_k = A_{n+1-k}^l p_k$ ,  $k=1,...,m+1$

Ta được bậc m và các hệ số  $d_k$ ,  $k=1,...,m+1$  của đa thức  $p^{(l)}(x)$ :

$$\begin{cases} m = n - l \\ a_k = \prod_{j=m+2-k}^{n+1-k} j, \quad k = 1, \dots, m+1 \\ d_k = a_k p_k \end{cases}$$

giá trị của đa thức  $p^{(l)}(x_0)$  được tính toán theo sơ đồ Horner.

Chương trình CT66 gồm việc vào dữ liệu các hệ số của đa thức, cấp của đạo hàm và lập một chu trình tính các hệ số của đạo hàm đa thức theo thuật toán tích tích một dãy số.

```
/* CT66 Chuong trinh dao ham da thuc */

#include <stdio.h>
#define MAXSIZE 21
typedef float vec[MAXSIZE];

void main()
{
    int i,j,k,l,n,np,mp;
    float ao;
    vec a,d;

    printf("\nChuong trinh tinh dao ham da thuc P(x)\n");
    printf("Bac cua da thuc P(x) n = ");
    scanf("%d",&n);
    printf("Cap cua dao ham l = ");
    scanf("%d",&l);
    printf("Cho cac he so cua da thuc\n");
```

```

for (i=0;i<=n;i++)
{
    printf("A[%d] = ",i);
    scanf("%f",&a[i]);
}
np = n+1;
mp = np-l;
for (k=1;k<=mp;k++)
{
    ao = 1;
    for (j=mp+1-k;j<=np-k;j++)
        ao = ao*j;
    d[k] = ao*a[k];
    printf("Dao ham cap %d D[%d] = %.6.2f\n",l,k,d[k]);
}
getch();
return;
}

```

Ta được kết quả chương trình CT66 như sau :

*Chương trình tính đạo hàm đa thức  $P(x)$*

*Bac cua da thuc  $P(x)$  n = 2*

*Cap cua dao ham l = 1*

*Cho cac he so cua da thuc*

*A[0] = 2*

*A[1] = 3*

*A[2] = 5*

*Dao ham cap 1 D[1] = 6.00*

*Dao ham cap 1 D[2] = 5.00*

**Bài số 67** Tính đạo hàm của đa thức Newton.

**Bài giải** Cho đa thức Newton

$$P(x) = a_1\Phi_n(x) + \dots + a_n\Phi_1(x) + a_{n+1}$$

đạo hàm cấp 1 của đa thức có thể được tính toán theo thuật toán CT66.

*/\* CT67 Chuong trinh dao ham da thuc Newton \*/*

```
#include <stdio.h>
#define MAXSIZE 21
typedef float vec[MAXSIZE];

void main()
{
    int i,k,l,np;
    float uo,xo,a1;
    vec a,b,z,x,y;
    printf("\nChuong trinh dao ham cap 1 da thuc Newton\n");
    printf(" Cho bac da thuc np = ");
    scanf("%d",&np);
    printf(" Cho cap cua dao ham l = ");
    scanf("%d",&l);
    printf(" Cho Xo = ");
    scanf("%f",&xo);
```

```

^ for (i=1;i<=np+1;i++)
{
    printf("X[%d] = ",i);
    scanf("%f",&x[i]);
    printf("Y[%d] = ",i);
    scanf("%f",&y[i]);
}

for (i=1;i<=np+1;i++)
    a[i] = y[i];
for (k=1;k<=np-1;k++)
{
    a1 = a[1];
    for (i=1;i<=np-k;i++)
    {
        a[i] = (a[i+1]-a[i])/(x[i+k]-x[i]);
        a[np-k+1] = a1;
        printf("A[%d] = %6.2f\n",i,a[i]);
    }
}
for (i=1;i<=np+1;i++)
    b[i] = a[i];
for (i=0;i<=l;i++)
    for (k=2;k<=np-i;k++)
        b[k] = b[k-1]*(xo-x[np+1-k-i]+b[k]);
uo = 1;
for (i=1;i<=l;i++)
{
    uo = uo*i;
}

```

```
z[i] = u0*b[np-i];
printf("Z[%d] = %6.2f\n",i,z[i]);
}
getch();
return;
}
```

Kết quả ta được :

*Chuong trinh dao ham cap l da thuc Newton*

*Cho bac da thuc np = 2*

*Cho cap cua dao ham l = 1*

*Cho X0 = 2*

*X[1] = 3*

*Y[1] = 4*

*X[2] = 5*

*Y[2] = 1*

*X[3] = 6*

*Y[3] = 2*

*M[1] = -1.50*

*Z[1] = -1.50*