

ÔN TẬP KỸ THUẬT LẬP TRÌNH

Phần 1: CÁC THUẬT TOÁN THÔNG DỤNG

A. Mảng một chiều:

- Khai báo: <kiểu dữ liệu> <tên mảng>[<số phần tử>];
- Truy cập vào một phần tử: <tên mảng>[<chỉ số>]
- Nhập dữ liệu cho mảng một chiều các số nguyên:

```
void Nhapmang(int a[],int n)
{
    for (int i=0;i<n;i++)
    {
        printf("Nhap a[%d] = ",i);
        scanf("%d",&a[i]);
    }
}
```

- Xuất dữ liệu cho mảng một chiều các số nguyên:

```
void Xuatmang(int a[],int n)
{
    for (int i=0;i<n;i++) printf("%d\t",a[i]);
}
```

- Tính tổng các phần tử của mảng:

```
int Tong(int a[], int n)
{
    int t=0;
    for(int i=0; i<n; i++) t=t+a[i];
    return t;
}
```

- Tính TBC các phần tử trong mảng:

```
float TBC(int a[], int n)
{
    int tong=0;
    for (int i=0; i<n; i++) tong+=a[i];
    return (float)tong/n; //ep kieu
}
```

- Tìm và in ra màn hình các phần tử chẵn trong mảng.

```
void PhanTuChan(int a[], int n)
{
    printf("\nDanh sach cac phan tu chan trong mang la:");
    for(int i=0; i<n; i++)
        if(a[i] % 2 == 0) printf("%3d",a[i]);
}
```

- Tìm và in ra màn hình các phần tử chẵn dương trong mảng.

```
void PhanTuChanDuong(int a[], int n)
```

```
{
    printf("\nDanh sach cac phan tu chan duong trong mang la:");
    for(int i=0; i<n; i++)
        if(a[i] % 2 == 0 && a[i] > 0) printf("%3d",a[i]);
}
```

- Tìm và in ra màn hình các phần tử là số nguyên tố có trong mảng.

```
int ktNguyenTo(int x)
{
    int uoc=0;
    for(int i=1; i<=x; i++)
        if (x%i==0) uoc++;
    if (uoc==2) return 1;
    return 0;
}
void inNguyenTo(int a[], int n)
{
    printf("\nDanh sach cac phan tu chan duong trong mang la:");
    for(int i=0; i<n; i++)
        if(ktNguyenTo(a[i] == 1) printf("%3d",a[i]);
}
```

- Tìm phần tử lớn nhất có trong mảng.

```
void TimMax(int a[], int n)
{
    int max = a [0];
    for(int i = 0; i < n ; i++)
        if(a[i] > max) max = a[i];
    printf("\nPhan tu lon nhat trong mang la: %d",max);
}
```

- Tìm phần tử nhỏ nhất có trong mảng.

```
void TimMin(int a[], int n)
{
    int min = a [0];
    for(int i = 0; i < n ; i++)
        if(a[i] < min) min = a[i];
    printf("\nPhan tu lon nhat trong mang la: %d",max);
}
```

- Sắp xếp tăng dần:

```
void HoanVi(int &a, int &b)
{
    int t;
    t = a; a = b; b = t;
}
void SapXepTang(float a[], int n)
{
    for(int i = 0; i < n-1; i++)
        for(int j = i+1; j < n; j++)
```

```
        if(a[i] > a[j]) HoanVi(a[i],a[j]);
    }
```

B. Ma trận:

- Khai báo: <kiểu dữ liệu> <tên ma trận>[<số dòng tối đa>][<số cột tối đa>];
- Truy cập một phần tử: <tên ma trận>[<chỉ số dòng>][<chỉ số cột>]
- Nhập số dòng, số cột:

```
void Enter(int &m,int &n)
{
    do
    {
        printf("Nhap so dong cua ma tran: ");
        scanf("%d",&m);
        if (m<1||m>100) printf("Nhap sai. Moi nhap lai.\n");
    }
    while (m<1||m>100);
    do
    {
        printf("Nhap so cot cua ma tran: ");
        scanf("%d",&n);
        if (n<1||n>100) printf("Nhap sai. Moi nhap lai.\n");
    }
    while (n<1||n>100);
}
```

- Nhập ma trận:

```
void Read(int a[][100],int m,int n)
{
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
        {
            printf("\tNhap a[%d][%d] = ",i,j);
            scanf("%d",&a[i][j]);
        }
}
```

- Xuất ma trận:

```
void Display(int a[][100],int m,int n)
{
    for (int i=0;i<m;i++)
    {
        for (int j=0;j<n;j++) printf("%d\t",a[i][j]);
        printf("\n");
    }
}
```

- Tính tổng các phần tử trong ma trận:

```
long Sum(int a[][100],int m,int n)
```

```
{
    long s=0;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            s+=a[i][j];
    return s;
}
```

- Tổng các phần tử trên đường chéo chính:

```
long MainDiagonal(int a[][100],int m)
{
    long s=0;
    for (int i=0;i<m;i++) s+=a[i][i];
    return s;
}
```

- Tổng các phần tử trên đường chéo phụ:

```
long DiagonalSide(int a[][100],int m,int n)
{
    long s=0;
    for (int i=0;i<m;i++) s+=a[i][n-1-i];
    return s;
}
```

- TBC các số dương:

```
float TBC(int a[][100],int m,int n)
{
    int count=0,s=0;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            if (a[i][j]>0)
            {
                count++;
                s+=a[i][j];
            }
    return (float) (s/count);
}
```

- Đếm số phần tử bằng x (cho trước):

```
int Count(int a[][100],int m,int n,int x)
{
    int c=0;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            if (a[i][j]==x) c++;
    return c;
}
```

- Hàm **memset**: dùng để khởi tạo giá trị ban đầu cho các phần tử của mảng 1 chiều/mảng 2 chiều, trong thư viện <memory.h>. Cách dùng: **memset(<tên mảng/mảng 2 chiều>,<giá trị cần đặt>,sizeof(<tên mảng>));** Ví dụ: khởi tạo mảng a gồm 10 phần tử có giá trị 0: `int a[10]; memset(a,0,sizeof(a));`
- Xoay ma trận:

+ Xoay 90⁰ cùng chiều kim đồng hồ:

```
for (int k=0; k < n; k++)
{
    for (int j=0; j<m; j++) printf("%4d",a[m-1-j][k]);
    printf("\n");
}
```

+ Xoay 90⁰ ngược chiều kim đồng hồ:

```
for (int k = 0; k < n; k++)
{
    for (int j=0; j<m; j++) printf("%4d",a[j][n-1- k]);
    printf("\n");
}
```

+ Xoay 180⁰:

```
for (int k = 0; k < m; k++)
{
    for (int j=0; j<n; j++) printf("%4d",a[m-1-k][n-1-j]);
    printf("\n");
}
```

- Sắp xếp tăng dần:

```
void SapxepTangDan(int a[][100],int m,int n)
{
    for (int i=0;i<m*n;i++)
        for (int j=0;j<m*n;j++)
            if (a[i/n][i%n]>a[j/n][j%n]) Traodoi(a[i/n][i%n],a[j/n][j%n]);
}
```

C. Kí tự:

- Khai báo: `char <tên biến>;`
- Thư viện: <string.h>
- Nhập kí tự:
 - + C1: `scanf("%c",&<tên biến>);`
 - + C2: `getch(<tên biến>);`
 - + C3: `char ch=getche();`
- Xuất kí tự:
 - + C1: `putch(<tên biến>);` //có thể in ra màu khi dùng với `textcolor(<mã màu từ 1 – 15>);`
 - + C2: `printf("%c",<tên biến>);`
 - + C3: `putc(<tên biến>);`

D. Chuỗi (xâu) kí tự:

- Khai báo: `char <tên biến>[<số kí tự tối đa>];` //nhớ +1 cho NULL
- Gán hằng: `char <tên biến>[] = "<giá trị cần gán>";`

- Thư viện: <string.h>
- Nhập/xuất chuỗi dùng <stdio.h>
 - + Nhập: scanf(“%s”,&<tên biến>); //không nhận dấu space, tab.
 - + Xuất: printf(“%s”,<tên biến>);
- Nhập/xuất chuỗi dùng <string.h>
 - + Nhập: gets(<tên biến>); //phải xoá bộ đệm bằng fflush(stdin); hay fflush(stdout); trong <stdlib.h>.
 - + Xuất: puts(<tên biến>); //xuất xong tự xuống dòng.
- Các hàm xử lý chuỗi (trong <string.h>):
 - Nối chuỗi: strcat(<tên chuỗi 1>,<tên chuỗi 2>); //kết quả chứa trong chuỗi 1.
 - Tìm lần xuất hiện đầu tiên của kí tự <x> trong chuỗi <a>: strchr(a,x);
 - So sánh: strcmp(<chuỗi 1>,<chuỗi 2>); // bằng nhau: ==0, lớn hơn: >0, bé hơn: <0.
 - Sao chép: strcpy(<chuỗi đích>,<chuỗi nguồn>);
 - Sao chép 1 phần: strncpy(<chuỗi đích>,<chuỗi nguồn>,<số kí tự cần chép>);
 - Tìm kiếm: strstr(<chuỗi 1>,<chuỗi 2>); //tìm sự xuất hiện đầu của chuỗi 2 trong chuỗi 1.
 - Lấy chiều dài: strlen(<tên chuỗi>);
 - Chuỗi thường -> hoa:strupr(<tên chuỗi>);
 - Chuỗi hoa -> thường: strlwr(<tên chuỗi>);
 - Chuỗi sang số: atoi(<tên chuỗi>); atol(<tên chuỗi>); atof(<tên chuỗi>);
 - Đảo chuỗi: strrev(<tên chuỗi>);

E. Kiểu dữ liệu có cấu trúc:

- Là một kiểu dữ liệu do người dùng tự định nghĩa. Bao gồm nhiều thành phần (trường – field), mỗi trường có một kiểu dữ liệu khác nhau.
- Cú pháp định nghĩa:

```
struct <tên cấu trúc>
{
    <kiểu> <trường 1>;
    <kiểu> <trường 2>;
    ...
    <kiểu> <trường n>;
};
```

- Khai báo: typedef struct <tên struct> <tên kiểu mới>;
- Truy cập vào từng thành phần: <tên struct>.<tên trường>
- Với kiểu dữ liệu là số thực, khi nhập, ta không nhập trực tiếp mà phải thông qua biến tạm.
- Mảng cấu trúc: là một mảng có từng phần tử là 1 struct.
- Khi nhập dữ liệu cho trường là kiểu chuỗi, phải xoá bộ đệm bằng fflush(stdin); hay fflush(stdout);

F. Kiểu con trỏ:

- Khai báo: <kiểu dữ liệu> *<tên biến con trỏ>
- Gán địa chỉ : phép &<biến> → <biến>.
- Lấy giá trị: <biến> → *<biến>.
- Cấp phát/thu hồi: trong <alloc.h>
 - + <biến con trỏ> = (<tên kiểu>*) malloc(<kích thước>); / free(<biến con trỏ>);

- + <biến con trỏ> = (<tên kiểu>*) calloc(<kích thước>,sizeof(<kiểu>)); / free(<biến con trỏ>);
- + <biến con trỏ> = new <kiểu dữ liệu>[<kích thước>]; / delete [] <biến con trỏ>;
- Cấp phát thêm: <biến con trỏ> = realloc(<biến con trỏ>,<kích thước cần cấp phát lại>);
- Con trỏ và mảng 1 chiều:
 - + &<tên mảng>[0] → <tên mảng>.
 - + &<tên mảng>[<vị trí>] → (<tên mảng> + <vị trí>).
 - + <tên mảng>[<vị trí>] → *(<tên mảng> + <vị trí>).

Hàm cấp phát bộ nhớ:

```
void Init(int *p,int n)
{
    p=(int*) calloc(n,sizeof(int));
    if (p==NULL)
    {
        printf("Khong du bo nho.\n");
        getch(); exit(1);
    }
}
```

- Con trỏ và ma trận:

Cách 1: con trỏ đơn cấp: *<tên biến con trỏ>.

- + <tên ma trận>[<chỉ số hàng>][<chỉ số cột>] → *(<tên ma trận> + <chỉ số hàng>*<số cột> + <chỉ số cột>).
- + &<tên ma trận>[<chỉ số hàng>][<chỉ số cột>] → (<tên ma trận> + <chỉ số hàng>*<số cột> + <chỉ số cột>).

Hàm cấp phát bộ nhớ:

```
void Init(int *p,int m,int n)
{
    p=(int*) calloc(m*n,sizeof(int));
    if (p==NULL)
    {
        printf("Khong du bo nho.\n");
        getch(); exit(1);
    }
}
```

Cách 2: con trỏ đa cấp: **<tên biến con trỏ>.

- + <tên ma trận>[<chỉ số hàng>][<chỉ số cột>] → *((<tên ma trận> + <chỉ số hàng>) + <chỉ số cột>).
- + &<tên ma trận>[<chỉ số hàng>][<chỉ số cột>] → *((<tên ma trận> + <chỉ số hàng>) + <chỉ số cột>).

Hàm cấp phát bộ nhớ:

```
void Init(int **a,int m,int n)
{
    a=(int**) calloc(m,sizeof(int*));
    if (a==NULL)
    {
        printf("Khong du bo nho.\n");
    }
}
```

```

        getch(); exit(1);
    }
    for (int i=0;i<m;i++)
    {
        a[i]=(int*) calloc(n,sizeof(int));
        if (a[i]==NULL)
        {
            printf("Khong du bo nho.\n");
            getch(); exit(1);
        }
    }
}

```

Hàm giải phóng bộ nhớ:

```

void MyFree(int **a,int m)
{
    for (int i=0;i<m;i++) free(a[i]);
    free(a);
}

```

- Con trỏ và kiểu dữ liệu có cấu trúc:

+ <tên struct>.<tên trường> ➔ <tên con trỏ struct> -> <tên trường> hoặc (*<tên con trỏ struct>).<tên trường>.

+ &<tên struct>.<tên trường> ➔ &<biến con trỏ struct> -> <tên trường>.

+ Truyền structure sang hàm: với hàm nhập, khi gọi tên hàm phải thêm &: <tên hàm>(&<tham số>,...);

G. Đề quy:

- Đề quy tuyến tính:

```

void <tên hàm>
{
    if <điều kiện dừng> return <giá trị hay kết thúc>; //phần neo
    else
    {
        //một số công việc
        //gọi đệ quy đến <tên hàm> //phần đệ quy
    }
}

```

- Đề quy nhị phân:

```

void <tên hàm>
{
    if <điều kiện dừng> return <giá trị hay kết thúc>; //phần neo
    else
    {
        //làm một số công việc
        //gọi đệ quy đến <tên hàm> để giải quyết vấn đề nhỏ hơn //phần đệ quy
        // gọi đệ quy đến <tên hàm> để giải quyết các vấn đề còn lại //phần đệ quy
    }
}

```


}

- **Đệ quy phi tuyến:**

void <tên hàm>

```
{
    for (int i=1;i<=n;i++)
    {
        //làm một số công việc
        if <điều kiện dừng> return <giá trị hay kết thúc>; //phần neo
        else //gọi đệ quy đến <tên hàm> //phần đệ quy
    }
}
```

- Đệ quy hỗ trợ **→** bên trong hàm này có lời gọi hàm kia.

- Giải toán bằng đệ quy:

+ Thông số hoá bài toán.

+ Tìm phần neo.

+ Tìm giải thuật gọi đệ quy lui dần về phần neo.

- Với các giải thuật đệ quy trên mảng, ta **giảm dần** số phần tử của mảng (<mảng>[0->n-1 phần tử], phân tích thành [0;n-2 phần tử], tìm phần neo với [n-1] và khi n={0,1}).

- Sắp xếp mảng 1 chiều bằng đệ quy:

Thuật toán QuickSort (sắp xếp nhanh):

```
#define Swap(type,a,b) {type tmp=a; a=b; b=tmp;}
void QuickSort(int a[],int l,int r)
{
    int key = a[(l+r)/2];
    int i=l,j=r;
    while(i <= j)
    {
        while(a[i]<key) i++;
        while(a[j]>key) j--;
        if(i <= j)
        {
            if (i<j) Swap(int,a[i],a[j]);
            i++; j--;
        }
    }
    if (l<j) QuickSort(a,l,j);
    if (i<r) QuickSort(a,i,r);
}
```

→ Lời gọi hàm: QuickSort(<tên mảng>,0,<số phần tử>);

H. Tập tin (File): (xét với tập tin văn bản)

- Khai báo: FILE *<tên biến tập tin>;

- Mở tập tin: <tên biến tập tin> = fopen("<đường dẫn>", "<chế độ>");

→ Chế độ:

Chế độ	Ý nghĩa
r	Mở tập tin văn bản để đọc
w	Tạo ra tập tin văn bản mới để ghi
a	Nối vào tập tin văn bản
r+	Mở một tập tin văn bản để đọc/ghi
w+	Tạo ra tập tin văn bản để đọc ghi
a+	Nối vào hay tạo mới tập tin văn bản để đọc/ghi

- Đóng tệp tin: `fclose(<tên biến tệp tin>);` ➔ `fcloseall();` //đóng tất cả các tệp tin.
- Kiểm tra đến cuối tệp: `feof(<tên biến tệp tin>);` //==EOF nếu cuối tệp, ngược lại ==0.
- Di chuyển con trỏ về đầu tệp: `rewind(<tên biến tệp tin>);`
- Đọc dữ liệu từ tệp tin:
 - + `getc(<tên biến tệp tin>);` //trả về mã ASCII của kí tự trong tệp tin liên kết với biến con trỏ.
 - + `fgets(<biến đích>,<độ dài chuỗi>,<tên biến tệp tin>);`
 - + `fscanf(<tên biến tệp tin>,"<định dạng>",<danh sách biến>);`
- Ghi dữ liệu lên tệp tin:
 - + `putc(<tên biến kí tự>,<tên biến tệp tin>);`
 - + `puts(<tên biến chuỗi>,<tên biến tệp tin>);`
 - + `fprintf(<tên biến tệp tin>,"<định dạng>",<danh sách biến>);`
- Quy trình làm việc với tệp tin:
Khai báo ➔ Mở file ➔ Đọc/ghi ➔ Đóng file.

Phần 2: CÁC BÀI ÔN TẬP THEO CHỦ ĐỀ

➔ Cấu trúc sinh viên:

```
#include"stdio.h"
#include"conio.h"
#include"string.h"
#include"stdlib.h"
struct sinhvien
{
    char masv[10];
    char hoten[30];
    int namsinh;
    float dtb;
};
void nhap1sv(sinhvien &a)
{
    printf("\nnhap ma sv: ");
    fflush();
    gets(a.masv);
    printf("\nnhap ho ten sv: ");
    fflush();
```

```
    gets(a.hoten);
    printf("\nnhap nam sinh: ");
    scanf("%d",&a.namsinh);
    printf("\nnhap diem trung binh: ");
    float d;
    scanf("%f",&d);
    a.dtb=d;
}

void nhapmangsv(sinhvien sv[], int &n)
{
    printf("\nnhap so luong sv: ");
    scanf("%d", &n);
    for(int i=0; i<n;i++)
    {
        printf("\nnhap sv thu %d", i);
        nhap1sv(sv[i]);
    }
}

void xuat1sv(sinhvien a)
{
    printf("\nma sv: ");
    puts(a.masv);
    printf("\nho ten: ");
    puts(a.hoten);
    printf("\nnam sinh %d",a.namsinh);
    printf("\ndiem trung binh %.1f",a.dtb);
}

void xuatmangsv(sinhvien sv[], int n)
{
    printf("\nthong tin sv da nhap");
    for(int i=0; i<n; i++)
    {
        printf("\nsv thu %d la: ", i);
        xuat1sv(sv[i]);
        printf("\n-----");
    }
}

void main()
{
    clrscr();
    sinhvien sv[20];
    int n;
    nhapmangsv(sv, n);
    xuatmangsv(sv,n);
    getch();
}
```

→ Cấu trúc phân số:

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

```
struct PS
{
    int tu;
    int mau;
};

void Enter(PS &x)
{
    printf("\tNhap tu so: ");
    scanf("%d",&x.tu);
    printf("\tNhap mau so: ");
    scanf("%d",&x.mau);
}

void Read(PS a[],int n)
{
    for (int i=0;i<n;i++)
    {
        Enter(a[i]);
        printf("-----\n");
    }
}

void Output(PS x)
{
    printf("%d/%d",x.tu,x.mau);
}

void Display(PS a[],int n)
{
    for (int i=0;i<n;i++)
    {
        Output(a[i]);
        printf("\t");
    }
}

int UCLN(int a,int b)
{
    while (b>0)
    {
        a=a%b;
        int tmp=a; a=b; b=tmp;
    }
    return a;
}

PS Rutgon(PS &x)
{
    int tmp=UCLN(x.tu,x.mau);
    x.tu=x.tu/tmp;
    x.mau=x.mau/tmp;
}
```

```

PS Cong(PS a,PS b)
{
    PS c;
    c.tu=a.tu*b.mau+a.mau*b.tu;
    c.mau=a.mau*b.mau;
    Rutgon(c);
    return c;
}

void Find(PS a[],int n)
{
    float b[100];
    memset(b,0,sizeof(b));
    for (int i=0;i<n;i++) b[i]=(float)a[i].tu/a[i].mau;
    float u=b[0],v=b[0]; int uu=0,vv=0;
    for (int i=1;i<n;i++)
    {
        if (b[i]>u)
        {
            u=b[i];
            uu=i;
        }
        if (b[i]<v)
        {
            v=b[i];
            vv=i;
        }
    }
    printf("\nPhan so lon nhat: ");
    Output(a[uu]);
    printf("\nPhan so nho nhat: ");
    Output(a[vv]);
}

int CountMax(PS a[],int n)
{
    float b[100];
    memset(b,0,sizeof(b));
    for (int i=0;i<n;i++) b[i]=(float)a[i].tu/a[i].mau;
    float tmp=b[0]; int res=0;
    for (int i=1;i<n;i++)
        if (b[i]>tmp) tmp=b[i];
    for (int i=0;i<n;i++)
        if (b[i]==tmp) res++;
    return res;
}

void main()
{
    clrscr();
    fflush(stdin);
    PS a[100];int n;
    printf("Nhap so luong phan so: ");
    scanf("%d",&n);

```

```

Read(a,n);
printf("Day phan so vua nhap:\n");
Display(a,n);
PS res;
res.tu=a[0].tu; res.mau=a[0].mau;
for (int i=1;i<n;i++) res=Cong(res,a[i]);
printf("\nTong cac phan so: ");
Output(res);
Find(a,n);
printf("\nSo phan tu lon nhat la: %d",CountMax(a,n));
getch();
}

```

→ Đề quy:

```

#include <stdio.h>
#include <conio.h>
#include <math.h>

#define Nm 100
#define Swap(type,a,b) {type tmp=a; a=b; b=tmp;}

void Enter(int &n)
{
    do
    {
        printf("NHap n = ");
        scanf("%d",&n);
        if (n<=0) printf("Nhap sai. Moi nhap lai.\n");
    }
    while (n<=0);
}

void EnterArray(int a[],int n)
{
    if (n>0)
    {
        EnterArray(a,n-1);
        printf("\tNhap a[%d] = ",n-1);
        scanf("%d",&a[n-1]);
    }
}

void Display(int a[],int n)
{
    if (n>0)
    {
        Display(a,n-1);
        printf("%d\t",a[n-1]);
    }
}

long EvenSum(int a[],int n)
{
    if (n==0) return 0;
    if (a[n-1]%2==0) return a[n-1]+EvenSum(a,n-1);
}

```

```
        return EvenSum(a,n-1);
    }

    int IsPrime(int x)
    {
        if (x<2) return 0;
        for (int i=2;i<=(int)sqrt(x);i++)
            if (x%i==0) return 0;
        return 1;
    }

    int PrimeCount(int a[],int n)
    {
        if (n==0) return 0; int c=0;
        if (IsPrime(a[n-1])==1) c++;
        return c+PrimeCount(a,n-1);
    }

    int FindMax(int a[],int n)
    {
        if (n==1) return a[0];
        if (a[n-1]>FindMax(a,n-1)) return a[n-1];
        return FindMax(a,n-1);
    }

    void QuickSort(int a[],int l,int r)
    {
        int key = a[(l+r)/2];
        int i=l,j=r;
        while(i <= j)
        {
            while(a[i]<key) i++;
            while(a[j]>key) j--;
            if(i <= j)
            {
                if (i<j) Swap(int,a[i],a[j]);
                i++; j--;
            }
        }
        if (l<j) QuickSort(a,l,j);
        if (i<r) QuickSort(a,i,r);
    }

    int IsSquare(int x)
    {
        return ((int)sqrt(x)==sqrt(x) ? 1:0);
    }

    void Square(int a[],int n)
    {
        if (n==0) return;
        if (IsSquare(a[n-1])==1) printf("%d\t",a[n-1]);
        Square(a,n-1);
    }
```

```

/* main program */

int main()
{
    clrscr();
    int a[Nm],n;
    Enter(n);
    EnterArray(a,n);
    printf("Mang vua nhap la:\n");
    Display(a,n);
    printf("\nTong cac so chan: %ld",EvenSum(a,n));
    printf("\nSo cac so nguyen to: %d",PrimeCount(a,n));
    printf("\nSo lon nhat la: %d",FindMax(a,n));
    QuickSort(a,0,n);
    printf("\nMang vua sap xep la:\n");
    Display(a,n);
    printf("\nCac so chinh phuong:\n");
    Square(a,n);
    getch();
    return 0;
}

```

→ Con trỏ với ma trận:

```

#include <stdio.h>
#include <conio.h>
#include <alloc.h>
#include <math.h>
#include <process.h>

void Enter(int &m,int &n)
{
    do
    {
        printf("Nhap so dong cua ma tran: ");
        scanf("%d",&m);
        if (m<1||m>100) printf("Nhap sai. Moi nhap lai.\n");
    }
    while (m<1||m>100);
    do
    {
        printf("Nhap so cot cua ma tran: ");
        scanf("%d",&n);
        if (n<1||n>100) printf("Nhap sai. Moi nhap lai.\n");
    }
    while (n<1||n>100);
}

void Init(int **a,int m,int n)
{
    a=(int**) calloc(m,sizeof(int*));
    if (a==NULL)
    {
        printf("Khong du bo nho.\n");
        getch(); exit(1);
    }
}

```



```
for (int i=0;i<m;i++)
{
    a[i]=(int*) calloc(n,sizeof(int));
    if (a[i]==NULL)
    {
        printf("Khong du bo nho.\n");
        getch(); exit(1);
    }
}

void MyFree(int **a,int m)
{
    for (int i=0;i<m;i++) free(a[i]);
    free(a);
}

void EnterArray(int **a,int m,int n)
{
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
        {
            printf("\tNhap a[%d][%d] = ",i,j);
            scanf("%d",&a[i][j]);
        }
}

void Display(int **a,int m,int n)
{
    for (int i=0;i<m;i++)
    {
        for (int j=0;j<n;j++) printf("%d\t",&a[i][j]);
        printf("\n");
    }
}

long ArraySum(int **a,int m,int n)
{
    long tmp=0;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++) tmp+=a[i][j];
    return tmp;
}

int IsPrime(int a)
{
    if (a<2) return 0;
    for (int i=2;i<=(int)sqrt(a);i++)
        if (a%i==0) return 0;
    return 1;
}

long PrimeSum(int **a,int m,int n)
{
    long tmp=0;
```

```
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            if (IsPrime(*(a+i+j))==1) tmp+=*(a+i+j);
    return tmp;
}

int MainDianoside(int **a,int m)
{
    int tmp=0;
    for (int i=0;i<m;i++) tmp+=*(a+i+i);
    return tmp;
}

int SideDianoside(int **a,int m)
{
    int tmp=0;
    for (int i=0;i<m;i++) tmp+=*(a+i+m-1-i);
    return tmp;
}

float Average(int **a,int m,int n)
{
    int tmp=0, count=0;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            if (*(a+i+j)>0)
            {
                tmp+=*(a+i+j);
                count++;
            }
    return (float)tmp/count;
}

int Count(int **a,int m,int n,int x)
{
    int count=0;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            if (*(a+i+j)==x) count++;
    return count;
}

int MaxArr(int **a,int m,int n)
{
    int tmp=**a;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            if (*(a+i+j)>tmp) tmp=*(a+i+j);
    return tmp;
}

int MinArr(int **a,int m,int n)
{
    int tmp=**a;
    for (int i=0;i<m;i++)
```

```

        for (int j=0;j<n;j++)
            if (*(a+i+j)<tmp) tmp=*(a+i+j);
    return tmp;
}

int MaxRow(int **a,int n,int i)
{
    int tmp=*(a+i);
    for (int j=0;j<n;j++)
        if (*(a+i+j)>tmp) tmp=*(a+i+j);
    return tmp;
}

int MinCollum(int **a,int m,int j)
{
    int tmp=*(a+j);
    for (int i=0;i<m;i++)
        if (*(a+i+j)<tmp) tmp=*(a+i+j);
    return tmp;
}

int MaxNegative(int **a,int m,int n)
{
    int tmp=1;
    for (int i=0;i<m;i++)
        for (int j=0;j<n;j++)
            if (*(a+i+j)<0)
            {
                tmp=*(a+i+j);
                break;
            }
    for (i=0;i<m;i++)
        for (j=0;j<n;j++)
            if (*(a+i+j)<0 && *(a+i+j)>tmp)
                tmp=*(a+i+j);
    return tmp;
}

int PrimeCount(int **a,int n,int i)
{
    int count=0;
    for (int j=0;j<n;j++)
        if (IsPrime(*(a+i+j))==1) count++;
    return count;
}

int PrimeRow(int **a,int m,int n)
{
    int res=-1;
    int tmp=0;
    for (int i=0;i<m;i++)
        if (PrimeCount(a,n,i)>tmp)
        {
            res=i;
            tmp=PrimeCount(a,n,i);
        }
}

```

```

    }
    return res;
}

/* main program */

int main()
{
    clrscr();
    int **a,m,n;
    Enter(m,n);
    Init(a,m,n);
    EnterArray(a,m,n);
    printf("Mang vua nhap la:\n");
    Display(a,m,n);
    printf("\nTong cac phan tu cua ma tran: %ld",ArraySum(a,m,n));
    printf("\nTong cac phan tu nguyen to cua ma tran: %ld",PrimeSum(a,m,n));
    if (m==n)
    {
        printf("\nTong cac phan tu tren duong cheo chinh: %d",MainDianoside(a,m));
        printf("\nTong cac phan tu tren duong cheo phu: %d",SideDianoside(a,m));
    }
    else printf("\nKhong co tong duong cheo chinh & phu.");
    printf("\nTBC cac so duong cua ma tran: %.1f",Average(a,m,n));
    int x;
    printf("\nNhap vao so can dem: ");
    scanf("%d",&x);
    printf("So lan xuat hien cua %d la: %d",x,Count(a,m,n,x));
    printf("\nPhan tu lon nhat trong mang: %d",MaxArr(a,m,n));
    printf("\nPhan tu nho nhat trong mang: %d",MinArr(a,m,n));
    printf("\nNhap dong can tim: ");
    scanf("%d",&x);
    printf("Phan tu lon nhat tren dong %d la: %d",x,MaxRow(a,n,x));
    printf("\nNhap cot can tim: ");
    scanf("%d",&x);
    printf("Phan tu nho nhat tren cot %d la: %d",x,MinCollum(a,m,x));
    if (MaxNegative(a,m,n)==1) printf("\nMang khong co so am.");
    else printf("\nPhan tu am co gia tri lon nhat: %d",MaxNegative(a,m,n));
    if (PrimeRow(a,m,n)==-1) printf("\nMang khong co SNT.");
    else printf("\nDong co nhieu SNT nhat: %d",PrimeRow(a,m,n));
    MyFree(a,m);
    getch();
    return 0;
}

```

→ Thao tác trên con trỏ:

```

#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#define ESC 27

void Caphpat(int *a,int n)
{
    a=new int[n];
    for(int i=0;i<n;i++)

```

```

        a[i]=0;
        return;
    }

void Mydelete(int *a)
{
    delete [] a;
    return;
}

void Nhap(int *a,int &n)
{
    do
    {
        printf("\nNhap so luong phan tu: ");
        scanf("%d", &n);
        if(n<1 || n>100 )
        {
            clrscr();
            printf("\nNhap sai! Nhap lai");
        }
    }while(n<1 || n>100);
    for(int i=0;i<n;i++)
    {
        printf("\nNhap phan tu thu a[%d]: ",i+1);
        scanf("%d", &a[i]);
    }
}

void Xuat(int *a,int n)
{
    for(int i=0;i<n;i++)
    {
        printf("%4d", a[i]);
    }
}

void Themdau(int *a,int &n)
{
    int x;
    printf("\n\tNhap phan tu muon them: ");
    scanf("%d", &x);
    for(int i=n;i>0;i--)
    {
        a[i]=a[i-1];
    }
    a[0]=x;
    n++;
    Capphat(a,n);
    printf("\nDa them phan tu %d vao dau mang.",x);
}

void Themvt(int *a,int &n)
{

```

```
int x,vt;
printf("\n\tNhap phan tu muon them: ");
scanf("%d", &x);
printf("\n\tNhap vi tri can them: ");
scanf("%d", &vt);
if(vt>=0 && vt<n)
{
    for(int i=n;i>vt;i--)
    {
        a[i]=a[i-1];
    }
    a[vt]=x;
    n++;
    Capphat(a,n);
    printf("\nDa them phan tu %d vao vi tri %d trong mang.",x,vt);
}
else
    printf("\nVi tri vuot qua gioi han!");
}

void Themcuoi(int *a,int &n)
{
    int x;
    printf("\n\tNhap phan tu muon them: ");
    scanf("%d", &x);
    a[n]=x;
    n++;
    Capphat(a,n);
    printf("\nDa them phan tu %d vao cuoi mang.",x);
}

void Xoadau(int *a,int &n)
{
    for(int i=0;i<n-1;i++)
        a[i]=a[i+1];
    n--;
    Capphat(a,n);
    printf("\n\tDa Xoa phan tu dau tien ra khoi mang.\n");
}

void Xoavt(int *a,int &n)
{
    int vt;
    printf("\n\tNhap vi tri muon xoa: ");
    scanf("%d", &vt);
    if(vt>=0 && vt<n)
    {
        for(int i=vt;i<n-1;i++)
            a[i]=a[i+1];
        n--;
        Capphat(a,n);
        printf("\nDa Xoa phan tu o vi tri %d ra khoi mang.\n",vt);
    }
    else
        printf("\nVi tri vuot qua gioi han!\n");
}
```

```
}

void Xoacuoai(int *a,int &n)
{
    a[n]=a[n-1];
    n--;
    Capphat(a,n);
    printf("\nDa Xoa phan tu cuoi cung ra khoi mang.\n");
}

int max(int *a,int n)
{
    int max=a[0];
    for(int i=0;i<n;i++)
        if(a[i]>max)
            max=a[i];
    return max;
}

void xoa(int *a,int &n,int k)
{
    for(int i=k;i<n-1;i++)
        a[i]=a[i+1];
    n--;
    Capphat(a,n);
}

void Xoamax(int *a,int &n)
{
    for(int i=0;i<n;i++)
    {
        if(a[i]==max(a,n))
        {
            xoa(a,n,i);
            i--;
        }
    }
    printf("\nDa xoa het phan tu max ra khoi mang.\n");
}

char menu()
{
    clrscr();
    printf("\n\t!^^MOT SO THAO TAC TREN MANG 1 CHIEU SU DUNG CON TRO^^!\n\n");
    printf("\tChon mot trong cac chuc nang:\n");
    printf("\t\t1. Nhap mang.\n");
    printf("\t\t2. Xuat mang.\n");
    printf("\t\t3. Them phan tu.\n");
    printf("\t\t4. Xoa phan tu.\n");
    printf("\n\tNhan ESC de thoat chuong trinh\n");
    return getch();
}

void Chuongtrinh()
{

```

```

clrscr();
int *a,n;
char ch;
do
{
    ch=menu();
    if(ch=='1')
    {
        Nhap(a,n);
        Capphat(a,n);
        getch();
    }
    if(ch=='2')
    {
        printf("\nDanh sach cac phan tu trong mang:\n\n");
        Xuat(a,n);
        getch();
    }
    if(ch=='3')
    {
        printf("\n\t\tTHEM PHAN TU VAO MANG\n\n");
        printf("\nChon mot chuc nang:\n");
        printf("\t\t3.1 Them dau.\n");
        printf("\t\t3.2 Them vao vi tri bat ki.\n");
        printf("\t\t3.3 Them cuoi.\n");
        char t=getch();
        if(t=='1')
            Themdau(a,n);
        if(t=='2')
            Themvt(a,n);
        if(t=='3')
            Themcuoi(a,n);
        getch();
    }
    if(ch=='4')
    {
        printf("\n\t\tXOA PHAN TU RA KHOI MANG\n\n");
        printf("\n\t\tChon mot trong cac chuc nang:\n");
        printf("\t\t4.1 Xoa dau.\n");
        printf("\t\t4.2 Xoa vi tri can xoa.\n");
        printf("\t\t4.3 Xoa cuoi.\n");
        printf("\t\t4.4 Xoa max.\n");
        char t=getch();
        if(t=='1')
            Xoadau(a,n);
        if(t=='2')
            Xoavt(a,n);
        if(t=='3')
            Xoacuai(a,n);
        if(t=='4')
            Xoamax(a,n);
        getch();
    }
} while(ch!=ESC);
Mydelete(a);

```



```

        getch();
    }

    void main()
    {
        clrscr();
        Chuongtrinh();
        getch();
    }

```

→ Một số hàm đệ quy:

```

#include <stdio.h>
#include <conio.h>
#include <math.h>
//-----
void nhap(int &n)
{
    do
    {
        printf("\n Nhap n: ");
        scanf("%d",&n);
        if(n<=0 || n>100) printf("\n Nhap sai. Nhap lai.");
    }while(n<=0 || n>100);
}
//-----
int nhapMang(int a[],int n)
{
    if(n == 0)
        return 0;
    nhapMang(a,n-1);
    printf("\n a[%d]= ",n-1);
    scanf("%d",&a[n-1]);
    return 0;
}
//-----
int xuấtMang(int a[],int n)
{
    if(n == 0)
        return 0;
    xuấtMang(a,n-1);
    printf("%4d",a[n-1]);
    return 0;
}
// xuất mang nguoc
int xuấtMangN(int a[],int n)
{
    if(n == 0)
        return 0;
    printf("%4d",a[n-1]);
    xuấtMang(a,n-1);
    //return 0;
}
//----- Tong -----
long tínhTong(int a[],int n)
{

```

```
        if(n == 0) return 0;
        long s = tinhTong(a,n-1);
        return s = s + a[n-1];
    }
    long tongChan(int a[],int n)
    {
        if(n == 0)
            return 0;
        long s=tongChan(a,n-1);
        if(a[n-1]%2 == 0) s = s + a[n-1];
        return s;
    }
    long tongLe(int a[],int n)
    {
        if(n == 0)
            return 0;
        long s = tongLe(a,n-1);
        if(a[n-1]%2 != 0) s = s + a[n-1];
        return s;
    }
    //----- So nguyen to -----
    int ktnt(int n)
    {
        int dem=0,i;
        for(i=1;i<=n;i++)
            if(n%i == 0) dem++;
        if(dem == 2) return 1;
        return 0;
    }
    long tongNT(int a[],int n)
    {
        if(n == 0)
            return 0;
        long s= tongNT(a,n-1);
        if(ktnt(a[n-1]) == 1) s = s + a[n-1];
        return s;
    }
    //----- tim le chan cuoi
    int leCuoi(int a[],int n)
    {
        if(n == 0)
            return 0;
        leCuoi(a,n-1);
        if(a[n-1]%2 != 0) return a[n-1];
    }
    int chanDau(int a[],int n)
    {
        if(n==0)return 1;
        int cd=chanDau(a,n-1);
        printf("\nchandau %d \t %d",cd,a[n-1]);
        if(cd!=1)return cd; else
            if(a[n-1]%2==0) return a[n-1];
        return 1;
    }
    //-----so chinh phuong
```

```

int ktra(int x)
{
    if(sqrt(x)==(int)sqrt(x)) return 1;
    return 0;
}
int cpCuoi(int a[],int n)
{
    if(n==0) return 0;
    cpCuoi(a,n-1);
    int tmp=a[n-1];
    if(tmp == 1) tmp=a[n-1];
    return tmp;
}
//----- tim Max
int timMax(int a[],int n)
{
    if(n==0) return 0;
    int t=timMax(a,n-1);
    if(t<a[n-1]) t=a[n-1];
    return t;
}
//----- dem Chan
int demChan(int a[],int n)
{
    int dem=0;
    if(n==0) return 0;
    dem=demChan(a,n-1);
    if(a[n-1]%2==0) dem=dem+1;
    return dem;
}
int demMax1(int a[],int n,int max)
{
    int dem;
    if(n==0) return 0;
    dem=demMax1(a,n-1,max);
    if(a[n-1]==max) dem++;
    return dem;
}
int vtMaxDau1(int a[],int n)
{
    if(n==0) return -1;
    if(n==1) return 0;
    int vt=vtMaxDau1(a,n-1);
    if(a[n-1] > a[vt]) return n-1;
    else return vt;
}
//----- sx tang mang
void hoanvi(int &a,int &b)
{
    int tmp;
    tmp=a; a=b; b=tmp;
}
void sxep(int a[],int n)
{
    if(n>1)

```

```

    {
        sxep(a,n-1);
        if(a[n-1]<a[n-2])
        {
            hoanvi(a[n-1],a[n-2]);
            sxep(a,n-1);
        }
    }
}

//-----
void main()
{
    clrscr();
    int a[100];int n,chon;
    printf("\n 1. Nhap mang 1 chieu.");
    printf("\n 2. Xuat mang 1 chieu.");
    printf("\n 3. Tinh tong mang.");
    printf("\n 4. Tinh tong cac phan tu chan.");
    printf("\n 5. Tinh tong cac phan tu le.");
    printf("\n 6. Tinh tong cac phan tu nguyen to trong mang.");
    printf("\n 7. Tim phan tu chinh phuong cuoi cung trong mang.");
    printf("\n 8. Phan tu lon nhat trong mang.");
    printf("\n 9. Dem so phan tu chan trong mang.");
    printf("\n 10. Sap xep mang tang dan.");
    printf("\n 11. Dem so phan tu lon nhat.");
    do
    {
        printf("\n -----Nhap cong viec can lam: ");
        scanf("%d",&chon);
        switch(chon)
        {
            case 1:
            {
                nhap(n);
                nhapMang(a,n);
                break;
            }
            case 2:
            {
                printf("\n Mang sau khi nhap la: \n");
                xuatMang(a,n);
                break;
            }
            case 3:
            {
                printf("\n Tong mang: %ld",tinhTong(a,n));
                break;
            }
            case 4:
            {
                printf("\n Tong chan: %ld",tongChan(a,n));
                break;
            }
            case 5:

```

```
        {
            printf("\n Tong le: %ld",tongLe(a,n));
            break;
        }
    case 6:
    {
        long nt = tongNT(a,n);
        if(nt == 0) printf("\n Khong co so nguyen to trong mang");
        else printf("\n Tong so nguyen to trong mang: %ld",nt);
        break;
    }
    case 7:
    {
        int t1 = cpCuoi(a,n);
        if(t1 == 0) printf("\n Khong co so chinh phuong trong mang");
        else printf("\n Chinh phuong cuoi la: %ld",t1);
        break;
    }
    case 8:
    {
        printf("\n Phan tu lon nhat trong mang la: %d",timMax(a,n));
        break;
    }
    case 9:
    {
        int tmp=demChan(a,n);
        if(tmp==0) printf("\n Khong co phan tu chan trong mang.");
        else printf("\n So phan tu chan trong mang la: %d ",tmp);
        break;
    }
    case 10:
    {
        sxep(a,n);
        xuatMang(a,n);
        break;
    }
    case 11:
    {
        int max=timMax(a,n);
        printf("\nDem so ptln la: %d",demMax1(a,n,max));
        break;
    }
    default:
    {
        chon=0;
        break;
    }
}
} while(chon!=0);
getch();
}
```

---HẾT---

Phước Nguyễn ©, April 18, 2016

```

1  /*Cau 1:
2  Chuong trinh nhap vao mang so thuc
3  ->Sap xep day so theo thu tu tu lon den nho
4  ->In day so sau khi da sap xep !
5  */
6  #include <stdio.h>
7  #include <conio.h>
8
9  //Khai bao nguyen mau ham
10 void NhapMang(float a[], int &n);
11 void XuatMang(float a[], int n);
12 void HoanVi(float &a, float &b);
13 void SapXepTang(float a[], int n);
14
15 void main()
16 {
17     //Khai bao mang so thuc co 100 phan tu
18     float a[100];
19     int n;
20     clrscr();
21     //Goi ham nhap mang:
22     NhapMang(a,n);
23     //Xuat mang da nhap
24     printf("\nMang da nhap la:");
25     XuatMang(a,n);
26     //Sap xep mang da nhap
27     SapXepTang(a,n);
28     //Goi ham xuat mang sau khi da sap xep
29     printf("\nMang sau khi sap xep tang la:");
30     XuatMang(a,n);
31     getch();
32 }
33
34 //Ham nhap mang, n truyen theo kieu tham bien
35 void NhapMang(float a[], int &n)
36 {
37     int i;
38     printf("\nNhap n:");
39     scanf("%d",&n);
40     for(i = 0;i < n; i++)
41     {
42         printf("a[%d] = ",i);
43         scanf("%f",&a[i]);
44     }
45 }
46 //Ham xuat mang, n truyen theo kieu tham tri vi khong thay gioi gia tri cua bien n
47 void XuatMang(float a[], int n)
48 {
49     int i;
50     for(i = 0;i < n; i++)
51     {
52         printf("%5.2f",a[i]);
53     }
54 }
55
56 //Ham hoan vi 2 so thuc

```

```

57 void HoanVi(float &a, float &b)
58 {
59     float t;
60     t = a;
61     a = b;
62     b = t;
63 }
64 //Ham sap xep day so tang
65 void SapXepTang(float a[], int n)
66 {
67     int i,j;
68     for(i = 0; i < n-1; i++)
69         for(j = i+1; j < n; j++)
70             {
71                 if(a[i] > a[j]) //Neu so dung truoc lon hon so dung sau thi hoan vi
72                     HoanVi(a[i],a[j]);
73             }
74 }
75 /*****
76 *****/
77 /*Bai 2:
78 Chuong trinh sap xep 1 mang theo thu tang dan
79 sau khi loai bo cac phan tu trung nhau !
80 */
81 #include <stdio.h>
82 #include <conio.h>
83
84 //Khai bao nguyen mau ham
85 void NhapMang(int a[], int &n);
86 void XuatMang(int a[], int n);
87 void HoanVi(int &a, int &b);
88 void SapXepTang(int a[], int n);
89 void RemoveExist(int a[], int &n);
90
91 void main()
92 {
93     //Khai bao mang so nguyen co 100 phan tu
94     int a[100];
95     int n;
96     clrscr();
97     //Goi ham nhap mang:
98     NhapMang(a,n);
99     //Xuat mang da nhap
100    printf("\nMang da nhap la:");
101    XuatMang(a,n);
102    //Xoa cac phan tu trung nhau trong mang
103    RemoveExist(a,n);
104    //Goi ham xuat mang sau khi da xoa cac phan tu trung nhau
105    printf("\nMang sau khi xoa cac phan tu trung nhau:");
106    XuatMang(a,n);
107    //Sap xep mang da nhap
108    SapXepTang(a,n);
109    //Goi ham xuat mang sau khi da sap xep
110    printf("\nMang sau khi sap xep tang la:");
111    XuatMang(a,n);
112    getch();

```

```

113 }
114
115 //Ham nhap mang, n truyen theo kieu tham bien
116 void NhapMang(int a[], int &n)
117 {
118     int i;
119     printf("\nNhap n:");
120     scanf("%d",&n);
121     for(i = 0;i < n; i++)
122     {
123         printf("a[%d] = ",i);
124         scanf("%d",&a[i]);
125     }
126 }
127 //Ham xuat mang, n truyen theo kieu tham tri vi khong thay gioi gia tri cua bien n
128 void XuatMang(int a[], int n)
129 {
130     int i;
131     for(i = 0;i < n; i++)
132     {
133         printf("%3d",a[i]);
134     }
135 }
136
137 //Ham hoan vi 2 so nguyen
138 void HoanVi(int &a, int &b)
139 {
140     int t;
141     t = a;
142     a = b;
143     b = t;
144 }
145 //Ham sap xep day so tang
146 void SapXepTang(int a[], int n)
147 {
148     int i,j;
149     for(i = 0; i < n-1; i++)
150         for(j = i+1; j < n; j++)
151         {
152             if(a[i] > a[j]) //Neu so dung truoc lon hon so dung sau thi hoan vi
153                 HoanVi(a[i],a[j]);
154         }
155 }
156 //Ham loai bo cac phan tu trung nhau trong mang
157 void RemoveExist(int a[], int &n)
158 {
159     int i,j,k;
160     for(i = 0; i < n-1; i++)
161         for(j = i+1; j < n; j++)
162         {
163             if(a[i] == a[j]) //Neu co 2 phan tu bang nhau
164             {
165                 for(k = j; k < n-1; k++)
166                     a[k] = a[k+1];
167                 n--;
168                 j--;

```



```

169         }
170
171     }
172 }
173 /*****
174 *****/
175 /*Bai 3:
176 Chuong trinh nhap vao mot mang
177 -Tim so lon nhat trong mang
178 -Phan tu nho nhat trong mang
179 -Tinh tong cac phan tu trong mang
180 */
181
182 //Khai bao thu vien
183 #include <stdio.h>
184 #include <conio.h>
185
186 //Khai bao nguyen mau ham
187 void NhapMang(int a[], int &n);
188 void XuatMang(int a[], int n);
189 int TimMax(int a[], int n);
190 int TimMin(int a[], int n);
191 int TinhTong(int a[], int n);
192
193 void main()
194 {
195     int a[100];
196     int n;
197     clrscr();
198     NhapMang(a,n);
199     printf("\nMang da nhap la:");
200     XuatMang(a,n);
201     printf("\nPhan tu lon nhat trong mang la: %d",TimMax(a,n));
202     printf("\nPhan tu nho nhat trong mang la: %d",TimMin(a,n));
203     getch();
204 }
205
206 //Ham nhap mang, n truyen theo kieu tham bien
207 void NhapMang(int a[], int &n)
208 {
209     int i;
210     printf("\nNhap n:");
211     scanf("%d",&n);
212     for(i = 0;i < n; i++)
213     {
214         printf("a[%d] = ",i);
215         scanf("%d",&a[i]);
216     }
217 }
218 //Ham xuat mang, n truyen theo kieu tham tri vi khong thay doi gia tri cua bien n
219 void XuatMang(int a[], int n)
220 {
221     int i;
222     for(i = 0;i < n; i++)
223     {
224         printf("%3d",a[i]);

```

```

225     }
226 }
227 //Ham tim phan tu lon nhat trong mang
228 int TimMax(int a[], int n)
229 {
230     int max = a[0]; //Gia su max la pt dau tien trong mang
231     int i;
232     for(i = 1; i < n; i++)
233         if(max < a[i])
234             max = a[i];
235     return max;
236 }
237 //Ham tim phan tu nho nhat trong mang
238 int TimMin(int a[], int n)
239 {
240     int min = a[0]; //Gia su min la pt dau tien trong mang
241     int i;
242     for(i = 1; i < n; i++)
243         if(min > a[i])
244             min = a[i];
245     return min;
246 }
247 /*****
248 *****/
249 /*Bai 4:
250 Chuong trinh nhap vao 1 mang tang
251 - Neu nguoi dung nhap sai quy tac thi nhap lai !
252 - Chen them 1 phan tu vao trong mang sao cho van dam bao nguyen tac: mang tang
253 */
254 //Khai bao thu vien
255 #include <stdio.h>
256 #include <conio.h>
257
258 //Khai bao nguyen mau ham
259 void NhapMang(int a[], int &n);
260 void XuatMang(int a[], int n);
261 int KiemTraMangTang(int a[], int n);
262 void ChenPhanTu(int a[], int &n, int x);
263
264 void main()
265 {
266     int a[100];
267     int n;
268     int x; //luu gia tri can chen vao trong mang
269     clrscr();
270     NhapMang(a, n);
271     while(KiemTraMangTang(a, n) == -1)
272     {
273         printf("\nNhap sai nguyen tac ! Nhap lai.");
274         getch();
275         clrscr();
276         NhapMang(a, n);
277     }
278     printf("\nMang da nhap la:");
279     XuatMang(a, n);
280     printf("\nNhap gia tri muon chen:");

```

```

281     scanf("%d",&x);
282     ChenPhanTu(a,n,x);
283     printf("\nMang sau khi chen phan tu %d vao mang la:",x);
284     XuatMang(a,n);
285     getch();
286 }
287 //Ham nhap mang, n truyen theo kieu tham bien
288 void NhapMang(int a[], int &n)
289 {
290     int i;
291     printf("\nNhap n:");
292     scanf("%d",&n);
293     for(i = 0;i < n; i++)
294     {
295         printf("a[%d] = ",i);
296         scanf("%d",&a[i]);
297     }
298 }
299 //Ham xuat mang, n truyen theo kieu tham tri vi khong thay doi gia tri cua bien n
300 void XuatMang(int a[], int n)
301 {
302     int i;
303     for(i = 0;i < n; i++)
304     {
305         printf("%3d",a[i]);
306     }
307 }
308 //Hamkiem tra mang tang
309 //Tra ve 0 neu mang tang
310 //Tra ve -1 neu mang khong tang
311 int KiemTraMangTang(int a[], int n)
312 {
313     int i;
314     for(i = 0;i < n-1; i++)//Dong for chay tu phan tu dau tien den pt gan cuoi
315         if(a[i] > a[i+1]) //Neu pt dung sau nho hon pt dung truoc
316             return -1; //Tra ve ma -1
317     return 0;
318 }
319 //Ham chen 1 gia tri vao mang sao cho van dam bao nguyen tac: mang tang
320 void ChenPhanTu(int a[], int &n, int x)
321 {
322     int i,j;
323     //Truong hop mang 0 co phan tu nao
324     if(n == 0)
325     {
326         a[0] = x;
327         n++;
328         return; //Thoat khoi chuong trinh
329     }
330     //Truong hop mang chi co duy nhathat mot phan tu
331     if(n == 1 && x > a[0])
332     {
333         a[1] =x;
334         n++;
335         return ; //Thoat khoi chuong trinh
336     }

```

```

337 //Cac truong hop con lai:
338 for(i = 0; i < n; i++)
339 {
340     if(x < a[i])
341     {
342         for(j = n-1; j >= i; j--) //Chay tu cuoi mang den vi tri i dang xet
343             a[j+1] = a[j];
344         //Sau khi day cac phan tu lon hon lai phia sau
345         //Chen x vao vi tri hien tai
346         a[i] = x;
347         n++;
348         break; //Thoat khoi vong lap
349     }
350 }
351 }
352 /*****
353 *****/
354 /*Bai 5:
355 Chuong trinh nhap vao mot ma tran (mang hai chieu) cac so nguyen.
356 Nhap mot so nguyen khac vao va xe xem co phan tu nao cua ma tran trung
357 voi so nay hay ko? O vi tri nao ? Co bao nhieu phan tu !
358 */
359 #include <stdio.h>
360 #include <conio.h>
361
362 //Khai bao nguyen mau ham:
363 void NhapMaTran(int a[][100],int n,int m);
364 void XuatMaTran(int a[][100],int n,int m);
365 void KiemTraTonTai(int a[][100], int n, int m, int x);
366
367 void main()
368 {
369     int a[100][100]; //Khai bao ma tran co 100 dong, 100 cot
370     int n; //so dong
371     int m; //so cot
372     int x; //gia tri do nguoi dung nhap vao
373     clrscr();
374     printf("\nNhap vao so dong:");
375     scanf("%d",&n);
376     printf("\nNhap vao so cot:");
377     scanf("%d",&m);
378     //Goi ham nhap ma tran
379     NhapMaTran(a,n,m);
380     //Goi ham xuat ma tran
381     printf("\nMa tran da nhap la:");
382     XuatMaTran(a,n,m);
383     printf("\nNhap vao gia tri x can tim:");
384     scanf("%d",&x);
385     //Tim vi tri cua x trong ma tran:
386     KiemTraTonTai(a,n,m,x);
387     getch();
388 }
389 //Ham nhap ma tran
390 void NhapMaTran(int a[][100],int n,int m)
391 {
392     for(int i= 0; i < n; i++)

```

```

393         for(int j = 0; j < m; j++)
394         {
395             printf("\nx[%d][%d]=",i,j);
396             scanf("%d",&a[i][j]);
397         }
398
399     }
400     //Ham xuất ma tran
401     void XuatMaTran(int a[][100],int n,int m)
402     {
403         int i,j;
404         for(i= 0; i < n; i++)
405         {
406             printf("\n");
407             for(j = 0; j < m; j++)
408             {
409                 printf("%3d",a[i][j]) ;
410             }
411         }
412     }
413     //Ham kiểm tra 1 phần tử mỗi nhập vào đã có trong ma tran hay chưa ?
414     void KiemTraTonTai(int a[][100], int n, int m, int x)
415     {
416         int i,j;
417         int count = 0; //Số phần tử trùng với x
418         for(i = 0; i < n; i++)
419             for(j = 0; j < m; j++)
420             {
421                 if(a[i][j] == x) //Có phần tử trùng
422                 {
423                     count++; //Tăng biến đếm lên 1
424                     printf("\nVi trí của %d trong ma tran la: dong(%d) - cot(%d)",x,i,j);
425                 }
426             }
427         //Neu count != 0 có nghĩa là tìm thấy vị trí trùng của x trong ma tran
428         if(count == 0)
429             printf("\nKhông tìm thấy %d trong ma tran !");
430         else
431             printf("\nSố lần tìm thấy %d trong ma tran la: %d",x,count);
432     }
433     /*****
434     *****/
435     /*Bai 6a:
436     Chuong trình chuyển đổi vị trí từ dòng thành cột của ma tran 4*4
437     */
438     #include <stdio.h>
439     #include <conio.h>
440
441     void XuatMaTran(int a[][4],int n,int m);
442     void ChuyenVi(int a[][4],int n, int m);
443
444     void main()
445     {
446         int a[][4] = {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16}; //Khai báo mảng tương minh
447         clrscr();
448         printf("\nMa tran hien tai:");

```

```

449     XuatMaTran(a,4,4);
450     printf("\nMa tran chuyen vi la:");
451     //Goi ham chuyen vi
452     ChuyenVi(a,4,4);
453     XuatMaTran(a,4,4);
454     getch();
455 }
456
457 void XuatMaTran(int a[][4],int n,int m)
458 {
459     int i,j;
460     for(i= 0; i < n; i++)
461     {
462         printf("\n");
463         for(j = 0; j < m; j++)
464         {
465             printf("%3d",a[i][j]) ;
466         }
467     }
468 }
469
470 //Ham hoan vi 2 so nguyen
471 void HoanVi(int &a, int &b)
472 {
473     int t;
474     t = a;
475     a = b;
476     b = t;
477 }
478
479 void ChuyenVi(int a[][4],int n, int m)
480 {
481     int i,j;
482     for(i = 0; i < n; i++)
483         for(j = i+1; j < m; j++)
484         {
485             HoanVi(a[i][j],a[j][i]);
486         }
487 }
488
489 /*****
490 *****/
491 /*Bai 6b:
492 Chuong trinh chuyen doi vi tri tu dong thanh cot cua ma tran m*n
493 */
494 #include <stdio.h>
495 #include <conio.h>
496
497 void NhapMaTran(int a[][100],int n,int m);
498 void XuatMaTran(int a[][100],int n,int m);
499 void ChuyenVi(int a[][100],int &n, int &m);
500
501 void main()
502 {
503     int a[100][100]; //Khai bao ma tran a co toi da 100 dong 100 cot
504     int n; //So dong

```

```

505     int m;    //So cot
506     clrscr();
507     printf("\nNhap vao so dong:");
508     scanf("%d",&n);
509     printf("\nNhap so cot:");
510     scanf("%d",&m);
511     NhapMaTran(a,n,m);
512     printf("\nMa tran da nhap la:");
513     XuatMaTran(a,n,m);
514     printf("\nMa tran chuyen vi la:");
515     ChuyenVi(a,n,m);
516     XuatMaTran(a,n,m);
517     getch();
518 }
519
520 void NhapMaTran(int a[][100],int n,int m)
521 {
522     for(int i= 0; i < n; i++)
523         for(int j = 0; j < m; j++)
524             {
525                 printf("\nx[%d][%d]=",i,j);
526                 scanf("%d",&a[i][j]);
527             }
528
529 }
530 void XuatMaTran(int a[][100],int n,int m)
531 {
532     int i,j;
533     for(i= 0; i < n; i++)
534     {
535         printf("\n");
536         for(j = 0; j < m; j++)
537             {
538                 printf("%3d",a[i][j]) ;
539             }
540     }
541 }
542 //Ham hoan vi 2 so nguyen
543 void HoanVi(int &a, int &b)
544 {
545     int t;
546     t = a;
547     a = b;
548     b = t;
549 }
550
551 void ChuyenVi(int a[][100],int &n, int &m)//Sau khi chuyen vi: dong->cot,cot->dong
552 {
553     int x; //Chieu cua ma tran vuong
554     int i,j;
555     int t;
556     x = (n > m ? n : m); //x la chieu lon hon
557     for(i = 0; i < x; i++)
558         for(j = i+ 1; j < x; j++)
559             HoanVi(a[i][j], a[j][i]);
560     //Luc nay dong thanh cot va cot thanh dong

```

```

561     HoanVi(n,m);
562 }
563 /*****
564 *****/
565 /*Bai 7:
566 Chuong trinh nhap vao mot mang so tu nhien. Hay xuat ra man hinh:
567 - Dong 1: gom cac so le, tong cong co bao nhieu so le.
568 - Dong 2: gom cac so chan, tong cong co bao nhieu so chan.
569 - Dong 3: gom cac so nguyen to.
570 - Dong 4: gom cac so khong phai la so nguyen to.
571 */
572 #include <stdio.h>
573 #include <conio.h>
574
575 //Khai bao nguyen mau ham
576 void NhapMang(int a[], int &n);
577 void XuatMang(int a[], int n);
578 void HienThiLe(int a[], int n);
579 void HienThiChan(int a[], int n);
580 void LaNguyenTo(int a[], int n);
581 void KhongLaNguyenTo(int a[], int n);
582
583 void main()
584 {
585     //Khai bao mang so nguyen co 100 phan tu
586     int a[100];
587     int n;
588     clrscr();
589     NhapMang(a,n);
590     printf("\nMang da nhap la:");
591     XuatMang(a,n);
592     HienThiLe(a,n);
593     HienThiChan(a,n);
594     LaNguyenTo(a,n);
595     KhongLaNguyenTo(a,n);
596     getch();
597 }
598
599 //Ham nhap mang, n truyen theo kieu tham bien
600 void NhapMang(int a[], int &n)
601 {
602     int i;
603     printf("\nNhap n:");
604     scanf("%d",&n);
605     for(i = 0;i < n; i++)
606     {
607         printf("a[%d] = ",i);
608         scanf("%d",&a[i]);
609     }
610 }
611 //Ham xuat mang, n truyen theo kieu tham tri vi khong thay gioi gia tri cua bien n
612 void XuatMang(int a[], int n)
613 {
614     int i;
615     for(i = 0;i < n; i++)
616     {

```



```

617         printf("%3d",a[i]);
618     }
619 }
620
621 //Ham kiem tra chan le
622 //Tra ve 0: chan
623 //Tra ve 1: le
624 int ChanLe(int a)
625 {
626     if(a % 2 == 0)
627         return 0;
628     return 1;
629 }
630 //Ham dem so luong phan tu co gia tri le va in len man hinh
631 void HienThiLe(int a[], int n)
632 {
633     int i;
634     int le = 0; //dem so phan tu le trong mang
635     printf("\nSo le:");
636     for(i = 0; i < n; i++)
637         if(ChanLe(a[i]) == 1)
638         {
639             le++;
640             printf("%3d",a[i]);
641         }
642     printf("---->So phan tu le trong mang la: %d",le);
643 }
644 //Ham dem so luong phan tu co gia tri chan va in len man hinh
645 void HienThiChan(int a[], int n)
646 {
647     int i;
648     int chan = 0; //dem so phan tu le trong mang
649     printf("\nSo chan:");
650     for(i = 0; i < n; i++)
651         if(ChanLe(a[i]) == 0)
652         {
653             chan++;
654             printf("%3d",a[i]);
655         }
656     printf("---->So phan tu chan trong mang la: %d",chan);
657 }
658 //Ham kiem tra so nguyen to
659 //Tra ve 0: neu ko phai la so nguyen to
660 //Tra ve 1: neu la so nguyen to
661 int KTNguyenTo(int n)
662 {
663     int i;
664     if(n == 1) //1 khong phai la so nguyen to
665         return 0;
666     for(i = 2; i <= n/2; i++)
667         if(n % i == 0) //Neu chia het
668             return 0;
669     return 1; //la so nguyen to
670 }
671 //Ham in cac so nguyen to trong mang
672 void LaNguyenTo(int a[], int n)

```

```

673 {
674     int i;
675     printf("\nSo nguyen to trong mang la:");
676     for(i = 0; i < n; i++)
677         if(KTNguyenTo(a[i]) == 1)
678             printf("%3d",a[i]);
679 }
680 //Ham in cac so khong phai so nguyen to trong mang
681 void KhongLaNguyenTo(int a[], int n)
682 {
683     int i;
684     printf("\nSo khong phai nguyen to trong mang la:");
685     for(i = 0; i < n; i++)
686         if(KTNguyenTo(a[i]) == 0)
687             printf("%3d",a[i]);
688 }
689 /*****
690 *****/
691 /*Bai 8:
692 Chuong trinh tin tong binh phuong cua cac so am trong 1 mang cac so nguyen
693 */
694 #include <stdio.h>
695 #include <conio.h>
696 #include <math.h>
697
698 //Khai bao nguyen mau ham
699 void NhapMang(int a[], int &n);
700 void XuatMang(int a[], int n);
701 long TongAm(int a[], int n);
702
703 void main()
704 {
705     //Khai bao mang so nguyen co 100 phan tu
706     int a[100];
707     int n;
708     clrscr();
709     //Goi ham nhap mang:
710     NhapMang(a,n);
711     //Xuat mang da nhap
712     printf("\nMang da nhap la:");
713     XuatMang(a,n);
714     printf("\nTong binh phuong cua cac so am: %d",TongAm(a,n));
715     getch();
716 }
717
718 //Ham nhap mang, n truyen theo kieu tham bien
719 void NhapMang(int a[], int &n)
720 {
721     int i;
722     printf("\nNhap n:");
723     scanf("%d",&n);
724     for(i = 0; i < n; i++)
725     {
726         printf("a[%d] = ",i);
727         scanf("%d",&a[i]);
728     }

```

```

729 }
730 //Ham xuất mang, n truyền theo kiểu tham trị vì không thay đổi giá trị của biến n
731 void XuấtMang(int a[], int n)
732 {
733     int i;
734     for(i = 0; i < n; i++)
735     {
736         printf("%3d", a[i]);
737     }
738 }
739 //Ham tính tổng bình phương các số âm trong mảng
740 long TổngÂm(int a[], int n)
741 {
742     int i;
743     long tong = 0;    //Tổng bình phương của các số âm
744     for(i = 0; i < n; i++)
745         if(a[i] < 0)    //Nếu nó là số âm
746         {
747             tong += pow(a[i], 2);
748         }
749     return tong;
750 }
751 /*****
752 *****/
753 /*Bai 9:
754 Chương trình thực hiện việc đảo 1 mảng 1 chiều
755 */
756 #include <stdio.h>
757 #include <conio.h>
758
759 //Khai báo nguyên mẫu hàm
760 void NhậpMang(int a[], int &n);
761 void XuấtMang(int a[], int n);
762 void ĐảoMang(int a[], int n);
763
764 void main()
765 {
766     //Khai báo mảng số nguyên có 100 phần tử
767     int a[100];
768     int n;
769     clrscr();
770     //Gọi hàm nhập mảng:
771     NhậpMang(a, n);
772     //Xuất mảng đã nhập
773     printf("\nMảng đã nhập là:");
774     XuấtMang(a, n);
775     printf("\nMảng sau khi đảo là:");
776     ĐảoMang(a, n);
777     XuấtMang(a, n);
778     getch();
779 }
780
781 //Ham nhập mảng, n truyền theo kiểu tham biến
782 void NhậpMang(int a[], int &n)
783 {
784     int i;

```

```

785     printf("\nNhap n:");
786     scanf("%d",&n);
787     for(i = 0;i < n; i++)
788     {
789         printf("a[%d] = ",i);
790         scanf("%d",&a[i]);
791     }
792 }
793 //Ham xuất mảng, n truyền theo kiểu tham trị vì không thay đổi giá trị của biến n
794 void XuấtMang(int a[], int n)
795 {
796     int i;
797     for(i = 0;i < n; i++)
798     {
799         printf("%3d",a[i]);
800     }
801 }
802 //Ham hoán vị 2 số nguyên
803 void HoánVi(int &a, int &b)
804 {
805     int t;
806     t = a;
807     a = b;
808     b = t;
809 }
810 //Ham đảo ngược mảng 1 chiều
811 void ĐảoMang(int a[], int n)
812 {
813     int i;
814     for(i = 0; i < n/2; i++)
815         HoánVi(a[i], a[n-1-i]);
816 }
817 /*****
818 *****/
819 /*Bài 10:
820 Chương trình nhập vào 2 ma trận có n dòng và m cột
821 ->Thực hiện phép cộng hai ma trận đã nhập
822 ->Xuất ma trận cộng lên màn hình
823 */
824 #include <stdio.h>
825 #include <conio.h>
826 void NhậpMaTran(int a[][100],int n,int m)
827 {
828     for(int i= 0; i < n; i++)
829         for(int j = 0; j < m; j++)
830         {
831             printf("\nx[%d][%d]=",i,j);
832             scanf("%d",&a[i][j]);
833         }
834 }
835 }
836 void XuấtMaTran(int a[][100],int n,int m)
837 {
838     int i,j;
839     for(i= 0; i < n; i++)
840     {

```

```

841     printf("\n");
842     for(j = 0; j < m; j++)
843     {
844         printf("%3d",a[i][j]) ;
845     }
846 }
847 }
848 //Ham cong hai ma tran voi nhau
849 void CongMaTran(int a[][100],int b[][100],int c[][100], int n, int m)
850 {
851     int i,j;
852     for(i =0 ;i<n;i++)
853         for(j=0;j<m;j++)
854         {
855             c[i][j] = a[i][j] + b[i][j];
856         }
857 }
858 void main()
859 {
860     int a[100][100], b[100][100];
861     int c[100][100];
862     int n;//luu so dong
863     int m;//lu so cot
864     clrscr();
865     printf("\nNhap vao so dong:");
866     scanf("%d",&n);
867     printf("\nNhap so cot:");
868     scanf("%d",&m);
869     //Nhap cac phan tu cho ma tran a
870     printf("\nNhap vao ma tran a:");
871     NhapMaTran(a,n,m);
872     //Nhap cac phan tu cho ma tran b
873     printf("\nNhap vao ma tran b:");
874     NhapMaTran(b,n,m);
875
876     printf("\nMa tran a da nhap la:");
877     XuatMaTran(a,n,m);
878     printf("\nMa tran b da nhap la:");
879     XuatMaTran(b,n,m);
880     CongMaTran(a,b,c,n,m);
881     printf("\nMa tran cong la:");
882     XuatMaTran(c,n,m);
883     getch();
884 }
885
886

```

```

1  /*Bai 1:
2  Chuong trinh nhap vao 1 day so thuc.
3  ->Sap xep day tang sau do in ra man hinh
4  */
5  #include <stdio.h>
6  #include <conio.h>
7  #include <stdlib.h>
8
9  void NhapMang(float *pa, int *pn);
10 void XuatMang(float *pa, int *pn);
11 void HoanVi(float *pa, float *pb);
12 void SapXepTang(float *pa, int *pn);
13
14 void main()
15 {
16     float *pa; //Khai bao con tro pa chi den mang a
17     int *pn ; //Khai bao con tro pn chi den bien n
18     pa = (float*) calloc(100,sizeof(float)); //Khoi tao vung nho chua 100 bien float
19     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
20     clrscr();
21     //Goi ham nhap mang:
22     NhapMang(pa,pn);
23     //Xuat mang da nhap
24     printf("\nMang da nhap la:");
25     XuatMang(pa,pn);
26     //Sap xep mang da nhap
27     SapXepTang(pa,pn);
28     //Goi ham xuat mang sau khi da sap xep
29     printf("\nMang sau khi sap xep tang la:");
30     XuatMang(pa,pn);
31     getch();
32 }
33 //Ham nhap mang su dung con tro
34 void NhapMang(float* pa, int *pn)
35 {
36     int i;
37     printf("\nNhap vao so phan tu cua mang:");
38     scanf("%d",pn);
39
40     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
41     {
42         printf("a[%d] = ",i);
43         scanf("%f",pa+i); //luu gia tri tai dia chi thu i cua mang
44     }
45 }
46 //Ham xuat mang su dung con tro
47 void XuatMang(float *pa, int *pn)
48 {
49     int i;
50     for(i = 0; i < *pn; i++)
51     {
52         printf("%.3f",*(pa+i)); //xuat gia tri tai vi tri i cua mang
53     }
54 }
55 //Ham hoan vi 2 so thuc
56 void HoanVi(float *a, float *b)

```

```

57 {
58     float t;
59     t = *a; /*a: lay gia tri tai o nho ma con tro a dang tro toi
60     *a = *b; /*b: lay gia tri tai o nho ma con tro b dang tro toi
61     *b = t;
62 }
63 //Ham sap xep day so tang
64 void SapXepTang(float *pa, int *pn)
65 {
66     int i,j;
67     for(i = 0; i < (*pn - 1); i++)
68         for(j = i+1; j < *pn; j++)
69             {
70                 if(*(pa+i) > *(pa+j)) //Neu so dung truoc lon hon so dung sau thi hoan vi
71                     HoanVi(pa+i,pa+j);
72             }
73 }
74
75 /*****
76 *****/
77 /*Bai 2:
78 Chuong trinh sap xep 1 mang theo thu tang dan
79 sau khi loai bo cac phan tu trung nhau !
80 */
81 #include <stdio.h>
82 #include <conio.h>
83 #include <stdlib.h>
84
85 //Khai bao nguyen mau ham
86 void NhapMang(int *pa, int *pn);
87 void XuatMang(int *pa, int *pn);
88 void HoanVi(int *pa, int *pb);
89 void SapXepTang(int *pa, int *pn);
90 void RemoveExist(int *pa, int *pn);
91
92 void main()
93 {
94     int *pa; //Khai bao con tro pa chi den mang a
95     int *pn ; //Khai bao con tro pn chi den bien n
96     pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
97     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
98     clrscr();
99     //Goi ham nhap mang:
100     NhapMang(pa,pn);
101     //Xuat mang da nhap
102     printf("\nMang da nhap la:");
103     XuatMang(pa,pn);
104     //Xoa cac phan tu trung nhau trong mang
105     RemoveExist(pa,pn);
106     //Goi ham xuat mang sau khi da xoa cac phan tu trung nhau
107     printf("\nMang sau khi xoa cac phan tu trung nhau:");
108     XuatMang(pa,pn);
109     //Sap xep mang da nhap
110     SapXepTang(pa,pn);
111     //Goi ham xuat mang sau khi da sap xep
112     printf("\nMang sau khi sap xep tang la:");

```

```

113     XuatMang(pa,pn);
114     getch();
115 }
116 //Ham nhap mang su dung con tro
117 void NhapMang(int* pa, int *pn)
118 {
119     int i;
120     printf("\nNhap vao so phan tu cua mang:");
121     scanf("%d",pn);
122
123     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
124     {
125         printf("a[%d] = ",i);
126         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
127     }
128 }
129 //Ham xuat mang su dung con tro
130 void XuatMang(int *pa, int *pn)
131 {
132     int i;
133     for(i = 0; i < *pn; i++)
134     {
135         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
136     }
137 }
138 //Ham hoan vi 2 so nguyen
139 void HoanVi(int *a, int *b)
140 {
141     int t;
142     t = *a; //a: lay gia tri tai o nho ma con tro a dang tro toi
143     *a = *b; //b: lay gia tri tai o nho ma con tro b dang tro toi
144     *b = t;
145 }
146 //Ham sap xep day so tang
147 void SapXepTang(int *pa, int *pn)
148 {
149     int i,j;
150     for(i = 0; i < (*pn - 1); i++)
151         for(j = i+1; j < *pn; j++)
152         {
153             if(*(pa+i) > *(pa+j)) //Neu so dung truoc lon hon so dung sau thi hoan vi
154                 HoanVi(pa+i,pa+j);
155         }
156 }
157
158 //Ham loai bo cac phan tu trung nhau trong mang
159 void RemoveExist(int *pa, int *pn)
160 {
161     int i,j,k;
162     for(i = 0; i < *pn-1; i++)
163         for(j = i+1; j < *pn; j++)
164         {
165             if(*(pa+i) == *(pa+j)) //Neu co 2 phan tu bang nhau
166             {
167                 for(k = j; k < *pn-1; k++)
168                     *(pa+k) = *(pa+k+1);

```



```

169         (*pn)--;
170         j--;
171     }
172
173 }
174 }
175 /*****
176 *****/
177 /*Bai 3:
178 Chuong trinh nhap vao mot mang
179 -Tim so lon nhat trong mang
180 -Phan tu nho nhat trong mang
181 -Tinh tong cac phan tu trong mang
182 */
183
184 //Khai bao thu vien
185 #include <stdio.h>
186 #include <conio.h>
187 #include <stdlib.h>
188
189 //Khai bao nguyen mau ham
190 void NhapMang(int *pa, int *pn);
191 void XuatMang(int *pa, int *pn);
192 int TimMax(int *pa, int *pn);
193 int TimMin(int *pa, int *pn);
194 int TinhTong(int *pa, int *pn);
195
196 void main()
197 {
198     int *pa; //Khai bao con tro pa chi den mang a
199     int *pn ; //Khai bao con tro pn chi den bien n
200     pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
201     pn = (int*) malloc(sizeof(int));      //Khoi tao vung nho chua 1 bien int
202     clrscr();
203     //Goi ham nhap mang:
204     NhapMang(pa,pn);
205     //Xuat mang da nhap
206     printf("\nMang da nhap la:");
207     XuatMang(pa,pn);
208     printf("\nPhan tu lon nhat trong mang la: %d",TimMax(pa,pn));
209     printf("\nPhan tu nho nhat trong mang la: %d",TimMin(pa,pn));
210     getch();
211 }
212
213 //Ham nhap mang su dung con tro
214 void NhapMang(int* pa, int *pn)
215 {
216     int i;
217     printf("\nNhap vao so phan tu cua mang:");
218     scanf("%d",pn);
219
220     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
221     {
222         printf("a[%d] = ",i);
223         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
224     }

```

```

225 }
226 //Ham xuất mảng su dung con tro
227 void XuatMang(int *pa, int *pn)
228 {
229     int i;
230     for(i = 0; i < *pn; i++)
231     {
232         printf("%3d",*(pa+i)); //xuất giá trị tại vị trí i của mảng
233     }
234 }
235 //Ham tìm phần tử lớn nhất trong mảng
236 int TimMax(int *pa, int *pn)
237 {
238     int max = *pa+0;//Giá trị max là pt đầu tiên trong mảng
239     int i;
240     for(i = 1; i < *pn; i++)
241         if(max < *(pa+i))
242             max = *(pa+i);
243     return max;
244 }
245 //Ham tìm phần tử nhỏ nhất trong mảng
246 int TimMin(int *pa, int *pn)
247 {
248     int min = *pa+0;//Giá trị min là pt đầu tiên trong mảng
249     int i;
250     for(i = 1; i < *pn; i++)
251         if(min > *(pa+i))
252             min = *(pa+i);
253     return min;
254 }
255 /*****
256 *****/
257 /*Bai 4:
258 Chương trình nhập vào 1 mảng tang
259 - Nếu người dùng nhập sai quy tắc thì nhập lại !
260 - Thêm 1 phần tử vào trong mảng sao cho vẫn đảm bảo nguyên tắc: mảng tang
261 */
262 //Khai báo thư viện
263 #include <stdio.h>
264 #include <conio.h>
265 #include <stdlib.h>
266
267 //Khai báo nguyên mẫu hàm
268 void NhapMang(int *pa, int *pn);
269 void XuatMang(int *pa, int *pn);
270 int KiemTraMangTang(int *pa, int *pn);
271 void ChenPhanTu(int *pa, int *pn, int x);
272
273 void main()
274 {
275     int *pa; //Khai báo con trỏ pa chỉ đến mảng a
276     int *pn ; //Khai báo con trỏ pn chỉ đến biến n
277     int x; //lưu giá trị cần thêm vào trong mảng
278     pa = (int*) calloc(100,sizeof(int)); //Khởi tạo vùng nhớ chứa 100 biến int
279     pn = (int*) malloc(sizeof(int)); //Khởi tạo vùng nhớ chứa 1 biến int
280     clrscr();

```

```

281 //Goi ham nhap mang:
282 NhapMang(pa,pn);
283 while(KiemTraMangTang(pa,pn)==-1)
284 {
285     printf("\nNhap sai nguyen tac ! Nhap lai.");
286     getch();
287     clrscr();
288     NhapMang(pa,pn);
289 }
290 printf("\nMang da nhap la:");
291 XuatMang(pa,pn);
292 printf("\nNhap gia tri muon chen:");
293 scanf("%d",&x);
294 ChenPhanTu(pa,pn,x);
295 printf("\nMang sau khi chen phan tu %d vao mang la:",x);
296 XuatMang(pa,pn);
297 getch();
298 }
299 //Ham nhap mang su dung con tro
300 void NhapMang(int* pa, int *pn)
301 {
302     int i;
303     printf("\nNhap vao so phan tu cua mang:");
304     scanf("%d",pn);
305
306     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
307     {
308         printf("a[%d] = ",i);
309         scanf("%d",pa+i); //luu gia tri tai dia chi thu i cua mang
310     }
311 }
312 //Ham xuat mang su dung con tro
313 void XuatMang(int *pa, int *pn)
314 {
315     int i;
316     for(i = 0; i < *pn; i++)
317     {
318         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
319     }
320 }
321 //Ham kiem tra mang tang
322 //Tra ve 0 neu mang tang
323 //Tra ve -1 neu mang khong tang
324 int KiemTraMangTang(int* pa, int *pn)
325 {
326     int i;
327     for(i = 0; i < *pn-1; i++)//Dong for chay tu phan tu dau tien den pt gan cuoi
328         if(*(pa+i) > *(pa+i+1)) //Neu pt dung sau nho hon pt dung truoc
329             return -1; //Tra ve ma -1
330     return 0;
331 }
332 //Ham chen 1 gia tri vao mang sao cho van dam bao nguyen tac: mang tang
333 void ChenPhanTu(int* pa, int *pn, int x)
334 {
335     int i,j;
336     //Truong hop mang 0 co phan tu nao

```

```

337     if(*pn == 0)
338     {
339         *(pa+0) = x;
340         (*pn)++;
341         return; //Thoat khoi chuong trinh
342     }
343     //Truong hop mang chi co duy nhat mot phan tu
344     if(*pn == 1 && x > *(pa+0))
345     {
346         *(pa+1) =x;
347         (*pn)++;
348         return ; //Thoat khoi chuong trinh
349     }
350     //Cac truong hop con lai:
351     for(i = 0; i < *pn; i++)
352     {
353         if(x < *(pa+i))
354         {
355             for(j = *pn-1; j >= i; j--) //Chay tu cuoi mang den vi tri i dang xet
356                 *(pa+j+1) = *(pa+j);
357             //Sau khi day cac phan tu lon hon lai phia sau
358             //Chen x vao vi tri hien tai
359             *(pa+i) = x;
360             (*pn)++;
361             break; //Thoat khoi vong lap
362         }
363     }
364 }
365 /*****
366 *****/
367 /*Bai 5:
368 Chuong trinh nhap vao mot ma tran (mang hai chieu) cac so nguyen.
369 Nhap mot so nguyen khac vao va xe xem co phan tu nao cua ma tran trung
370 voi so nay hay ko? O vi tri nao ? Co bao nhieu phan tu !
371 */
372 #include <stdio.h>
373 #include <conio.h>
374 #include <stdlib.h>
375
376 //Khai bao nguyen mau ham:
377 void NhapMaTran(int *pa, int *pn,int *pm);
378 void XuatMaTran(int *pa, int *pn,int *pm);
379 void KiemTraTonTai(int *pa, int *pn,int *pm, int x);
380
381 void main()
382 {
383     int *pa; //Khai bao con tro pa chi den ma tran a
384     int *pn ; //Khai bao con tro pn chi den bien n(dong)
385     int *pm ; //Khai bao con tro pn chi den bien m(cot)
386     pa = (int*) calloc(100*100,sizeof(int)); //Khoi tao vung nho chua 10000 bien int
387     pn = (int*) malloc(sizeof(int));          //Khoi tao vung nho chua 1 bien int
388     pm = (int*) malloc(sizeof(int));          //Khoi tao vung nho chua 1 bien int
389     int x;  //gia tri do nguoi dung nhap vao
390     clrscr();
391     //Goi ham nhap ma tran
392     NhapMaTran(pa,pn,pm);

```

```

393     //Goi ham xuất ma tran
394     printf("\nMa tran da nhap la:");
395     XuatMaTran(pa,pn,pm);
396     printf("\nNhap vao gia tri x can tim:");
397     scanf("%d",&x);
398     //Tim vi tri cua x trong ma tran:
399     KiemTraTonTai(pa,pn,pm,x);
400     getch();
401 }
402 //Ham nhap ma tran su dung con tro
403 void NhapMaTran(int *pa, int *pn,int *pm)
404 {
405     printf("\nNhap vao so dong:");
406     scanf("%d",pn);    //pn chinh la dia chi cua bien n
407     printf("\nNhap vao so cot:");
408     scanf("%d",pm);    //pm chinh la dia chi cua bien m
409     for(int i= 0; i < *pn; i++)
410         for(int j = 0; j < *pm; j++)
411         {
412             printf("\nx[%d][%d]=",i,j);
413             scanf("%d",(pa+i*(*pm)+j));
414         }
415
416 }
417 //Ham xuất ma tran su dung con tro
418 void XuatMaTran(int *pa, int *pn,int *pm)
419 {
420     int i,j;
421     for(i= 0; i < *pn; i++)
422     {
423         printf("\n");
424         for(j = 0; j < *pm; j++)
425         {
426             printf("%3d",*(pa+i*(*pm)+j)) ;
427         }
428     }
429 }
430 //Ham kiem tra 1 phan tu moi nhap vao da co trong ma tran hay chua ?
431 void KiemTraTonTai(int *pa, int *pn,int *pm, int x)
432 {
433     int i,j;
434     int count = 0; //So phan tu trung voi x
435     for(i = 0; i < *pn; i++)
436         for(j = 0; j < *pm; j++)
437         {
438             if(*(pa+i*(*pm)+j) == x) //Co phan tu trung
439             {
440                 count++; //Tang bien dem len 1
441                 printf("\nTim thay %d tai vi tri: dong(%d) - cot(%d)",x,i,j);
442             }
443         }
444     //Neu count != 0 co nghia la tim thay vi tri trung cua x trong ma tran
445     if(count == 0)
446         printf("\nKhong tim thay %d trong ma tran !");
447     else
448         printf("\nSo lan tim thay %d trong ma tran la: %d",x,count);

```

```

449 }
450 /*****
451 *****/
452 /*Bai 6:
453 Chuong trinh chuyen doi vi tri tu dong thanh cot cua ma tran m*n
454 */
455 #include <stdio.h>
456 #include <conio.h>
457 #include <stdlib.h>
458
459 void NhapMaTran(int *pa, int *pn,int *pm);
460 void XuatMaTran(int *pa, int *pn,int *pm);
461 void ChuyenVi(int *pa, int *pn,int *pm);
462
463 void main()
464 {
465     int *pa; //Khai bao con tro pa chi den ma tran a
466     int *pn ; //Khai bao con tro pn chi den bien n(dong)
467     int *pm ; //Khai bao con tro pn chi den bien m(cot)
468     pa = (int*) calloc(100*100,sizeof(int)); //Khoi tao vung nho chua 10000 bien int
469     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
470     pm = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
471     clrscr();
472     NhapMaTran(pa,pn,pm);
473     printf("\nMa tran da nhap la:");
474     XuatMaTran(pa,pn,pm);
475     printf("\nMa tran chuyen vi la:");
476     ChuyenVi(pa,pn,pm);
477     XuatMaTran(pa,pn,pm);
478     getch();
479 }
480
481 //Ham nhap ma tran su dung con tro
482 void NhapMaTran(int *pa, int *pn,int *pm)
483 {
484     printf("\nNhap vao so dong:");
485     scanf("%d",pn); //pn chinh la dia chi cua bien n
486     printf("\nNhap vao so cot:");
487     scanf("%d",pm); //pm chinh la dia chi cua bien m
488     for(int i= 0; i < *pn; i++)
489         for(int j = 0; j < *pm; j++)
490             {
491                 printf("\nx[%d][%d]=",i,j);
492                 scanf("%d",(pa+i*(*pm)+j));
493             }
494
495 }
496 //Ham xuat ma tran su dung con tro
497 void XuatMaTran(int *pa, int *pn,int *pm)
498 {
499     int i,j;
500     for(i= 0; i < *pn; i++)
501     {
502         printf("\n");
503         for(j = 0; j < *pm; j++)
504             {

```

```

505         printf("%3d",*(pa+i*(*pm)+j)) ;
506     }
507 }
508 }
509
510 //Ham hoan vi 2 so nguyen
511 void HoanVi(int *a, int *b)
512 {
513     int t;
514     t = *a; //a: lay gia tri tai o nho ma con tro a dang tro toi
515     *a = *b; //b: lay gia tri tai o nho ma con tro b dang tro toi
516     *b = t;
517 }
518
519 void ChuyenVi(int *pa, int *pn, int *pm) //Sau khi chuyen vi: dong->cot, cot->dong
520 {
521     int x; //Chieu cua ma tran vuong
522     int i, j;
523     int t;
524     x = (*pn > *pm ? *pn : *pm); //x la chieu lon hon
525     for(i = 0; i < x; i++)
526         for(j = i + 1; j < x; j++)
527             HoanVi((pa+i*(*pm)+j), (pa+j*(*pm)+i));
528     //Luc nay dong thanh cot va cot thanh dong
529     HoanVi(pn, pm);
530 }
531 /*****
532 *****/
533 /*Bai 7:
534 Chuong trinh nhap vao mot mang so tu nhien. Hay xuat ra man hinh:
535 - Dong 1: gom cac so le, tong cong co bao nhieu so le.
536 - Dong 2: gom cac so chan, tong cong co bao nhieu so chan.
537 - Dong 3: gom cac so nguyen to.
538 - Dong 4: gom cac so khong phai la so nguyen to.
539 */
540 #include <stdio.h>
541 #include <conio.h>
542 #include <stdlib.h>
543
544 //Khai bao nguyen mau ham
545 void NhapMang(int *pa, int *pn);
546 void XuatMang(int *pa, int *pn);
547 void HienThiLe(int *pa, int *pn);
548 void HienThiChan(int *pa, int *pn);
549 void LaNguyenTo(int *pa, int *pn);
550 void KhongLaNguyenTo(int *pa, int *pn);
551
552 void main()
553 {
554     int *pa; //Khai bao con tro pa chi den mang a
555     int *pn; //Khai bao con tro pn chi den bien n
556     pa = (int*) calloc(100, sizeof(int)); //Khoi tao vung nho chua 100 bien int
557     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
558     clrscr();
559     //Goi ham nhap mang:
560     NhapMang(pa, pn);

```

```

561     //Xuat mang da nhap
562     printf("\nMang da nhap la:");
563     XuatMang(pa,pn);
564     HienThiLe(pa,pn);
565     HienThiChan(pa,pn);
566     LaNguyenTo(pa,pn);
567     KhongLaNguyenTo(pa,pn);
568     getch();
569 }
570
571 //Ham nhap mang su dung con tro
572 void NhapMang(int* pa, int *pn)
573 {
574     int i;
575     printf("\nNhap vao so phan tu cua mang:");
576     scanf("%d",pn);
577
578     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
579     {
580         printf("a[%d] = ",i);
581         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
582     }
583 }
584 //Ham xuat mang su dung con tro
585 void XuatMang(int *pa, int *pn)
586 {
587     int i;
588     for(i = 0; i < *pn; i++)
589     {
590         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
591     }
592 }
593
594 //Ham kiem tra chan le
595 //Tra ve 0: chan
596 //Tra ve 1: le
597 int ChanLe(int a)
598 {
599     if(a % 2 == 0)
600         return 0;
601     return 1;
602 }
603 //Ham dem so luong phan tu co gia tri le va in len man hinh
604 void HienThiLe(int *pa, int *pn)
605 {
606     int i;
607     int le = 0; //dem so phan tu le trong mang
608     printf("\nSo le:");
609     for(i = 0; i < *pn; i++)
610         if(ChanLe(*(pa+i)) == 1)
611         {
612             le++;
613             printf("%3d",*(pa+i));
614         }
615     printf("---->So phan tu le trong mang la: %d",le);
616 }

```



```

617 //Ham dem so luong phan tu co gia tri chan va in len man hinh
618 void HienThiChan(int *pa, int *pn)
619 {
620     int i;
621     int chan = 0; //dem so phan tu le trong mang
622     printf("\nSo chan:");
623     for(i = 0; i < *pn; i++)
624         if(ChanLe(*(pa+i)) == 0)
625         {
626             chan++;
627             printf("%3d",*(pa+i));
628         }
629     printf("--->So phan tu chan trong mang la: %d",chan);
630 }
631 //Ham kiem tra so nguyen to
632 //Tra ve 0: neu ko phai la so nguyen to
633 //Tra ve 1: neu la so nguyen to
634 int KTNguyenTo(int n)
635 {
636     int i;
637     if(n == 1) //1 khong phai la so nguyen to
638         return 0;
639     for(i = 2; i <= n/2; i++)
640         if(n % i == 0) //Neu chia het
641             return 0;
642     return 1; //la so nguyen to
643 }
644 //Ham in cac so nguyen to trong mang
645 void LaNguyenTo(int *pa, int *pn)
646 {
647     int i;
648     printf("\nSo nguyen to trong mang la:");
649     for(i = 0; i < *pn; i++)
650         if(KTNguyenTo(*(pa+i)) == 1)
651             printf("%3d",*(pa+i));
652 }
653 //Ham in cac so khong phai so nguyen to trong mang
654 void KhongLaNguyenTo(int *pa, int *pn)
655 {
656     int i;
657     printf("\nSo khong phai nguyen to trong mang la:");
658     for(i = 0; i < *pn; i++)
659         if(KTNguyenTo(*(pa+i)) == 0)
660             printf("%3d",*(pa+i));
661 }
662 /*****
663 *****/
664 /*Bai 8:
665 Chuong trinh tinh tong binh phuong cua cac so am trong 1 mang cac so nguyen
666 */
667 #include <stdio.h>
668 #include <conio.h>
669 #include <math.h>
670 #include <stdlib.h>
671
672 //Khai bao nguyen mau ham

```

```

673 void NhapMang(int *pa, int *pn);
674 void XuatMang(int *pa, int *pn);
675 long TongAm(int *pa, int *pn);
676
677 void main()
678 {
679     int *pa; //Khai bao con tro pa chi den mang a
680     int *pn ; //Khai bao con tro pn chi den bien n
681     pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
682     pn = (int*) malloc(sizeof(int));      //Khoi tao vung nho chua 1 bien int
683     clrscr();
684     //Goi ham nhap mang:
685     NhapMang(pa,pn);
686     //Xuat mang da nhap
687     printf("\nMang da nhap la:");
688     XuatMang(pa,pn);
689     printf("\nTong binh phuong cua cac so am: %d",TongAm(pa,pn));
690     getch();
691 }
692
693 //Ham nhap mang su dung con tro
694 void NhapMang(int* pa, int *pn)
695 {
696     int i;
697     printf("\nNhap vao so phan tu cua mang:");
698     scanf("%d",pn);
699
700     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
701     {
702         printf("a[%d] = ",i);
703         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
704     }
705 }
706 //Ham xuat mang su dung con tro
707 void XuatMang(int *pa, int *pn)
708 {
709     int i;
710     for(i = 0; i < *pn; i++)
711     {
712         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
713     }
714 }
715
716 //Ham tinh tong binh phuong cac so am trong mang
717 long TongAm(int* pa, int *pn)
718 {
719     int i;
720     long tong = 0;    //Tong binh phuong cua cac so am
721     for(i = 0; i < *pn; i++)
722         if(*(pa+i) < 0)    //Neu no la so am
723         {
724             tong += pow(*(pa+i),2);
725         }
726     return tong;
727 }
728 /*****

```

```

729  *****/
730  /*Bai 9:
731  Chuong trinh thuc hien viec dao 1 mang 1 chieu
732  */
733  #include <stdio.h>
734  #include <conio.h>
735  #include <stdlib.h>
736
737  //Khai bao nguyen mau ham
738  void NhapMang(int *pa, int *pn);
739  void XuatMang(int *pa, int *pn);
740  void DaoMang(int *pa, int *pn);
741
742  void main()
743  {
744      int *pa; //Khai bao con tro pa chi den mang a
745      int *pn ; //Khai bao con tro pn chi den bien n
746      pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
747      pn = (int*) malloc(sizeof(int));      //Khoi tao vung nho chua 1 bien int
748      clrscr();
749      //Goi ham nhap mang:
750      NhapMang(pa,pn);
751      //Xuat mang da nhap
752      printf("\nMang da nhap la:");
753      XuatMang(pa,pn);
754      printf("\nMang sau khi dao la:");
755      DaoMang(pa,pn);
756      XuatMang(pa,pn);
757      getch();
758  }
759
760  //Ham nhap mang su dung con tro
761  void NhapMang(int* pa, int *pn)
762  {
763      int i;
764      printf("\nNhap vao so phan tu cua mang:");
765      scanf("%d",pn);
766
767      for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
768      {
769          printf("a[%d] = ",i);
770          scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
771      }
772  }
773  //Ham xuat mang su dung con tro
774  void XuatMang(int *pa, int *pn)
775  {
776      int i;
777      for(i = 0; i < *pn; i++)
778      {
779          printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
780      }
781  }
782  //Ham hoan vi 2 so nguyen
783  void HoanVi(int *a, int *b)
784  {

```

```

785     int t;
786     t = *a; /*a: lay gia tri tai o nho ma con tro a dang tro toi
787     *a = *b; /*b: lay gia tri tai o nho ma con tro b dang tro toi
788     *b = t;
789 }
790 //Ham dao nguoc mang 1 chieu
791 void DaoMang(int *pa, int *pn)
792 {
793     int i;
794     for(i = 0; i < (*pn)/2; i++)
795         HoanVi((pa+i), (pa+(*pn)-1-i));
796 }
797 /*****
798 *****/
799 /*Bai 10:
800 Chuong trinh nhap vao 2 ma tran co n dong va m cot
801 ->Thuc hien phep cong hai ma tran da nhap
802 ->Xuat ma tran cong len man hinh
803 */
804 #include <stdio.h>
805 #include <conio.h>
806 #include <stdlib.h>
807
808 //Khai bao nguyen mau ham:
809 void NhapMaTran(int *pa, int *pn, int *pm);
810 void XuatMaTran(int *pa, int *pn, int *pm);
811 void CongMaTran(int *pa, int *pb, int *pc, int *pn, int *pm);
812
813
814 void main()
815 {
816     int *pa; //Khai bao con tro pa chi den ma tran a
817     int *pb; //Khai bao con tro pb chi den ma tran b
818     int *pc; //Khai bao con tro pc chi den ma tran c
819     int *pn; //Khai bao con tro pn chi den bien n(dong)
820     int *pm; //Khai bao con tro pm chi den bien m(cot)
821     pa = (int*) calloc(100*100, sizeof(int)); //Khoi tao vung nho chua 10000 bien int
822     pb = (int*) calloc(100*100, sizeof(int)); //Khoi tao vung nho chua 10000 bien int
823     pc = (int*) calloc(100*100, sizeof(int)); //Khoi tao vung nho chua 10000 bien int
824     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
825     pm = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
826
827     printf("\nNhap vao so dong:");
828     scanf("%d", pn); //pn chinh la dia chi cua bien n
829     printf("\nNhap vao so cot:");
830     scanf("%d", pm); //pm chinh la dia chi cua bien m
831
832     //Nhap cac phan tu cho ma tran a
833     printf("\nNhap vao ma tran a:");
834     NhapMaTran(pa, pn, pm);
835     //Nhap cac phan tu cho ma tran b
836     printf("\nNhap vao ma tran b:");
837     NhapMaTran(pb, pn, pm);
838
839     printf("\nMa tran a da nhap la:");
840     XuatMaTran(pa, pn, pm);

```

```

841     printf("\nMa tran b da nhap la:");
842     XuatMaTran(pb,pn,pm);
843     CongMaTran(pa,pb,pc,pn,pm);
844     printf("\nMa tran cong la:");
845     XuatMaTran(pc,pn,pm);
846     getch();
847 }
848 //Ham nhap ma tran su dung con tro
849 void NhapMaTran(int *pa, int *pn,int *pm)
850 {
851     for(int i= 0; i < *pn; i++)
852         for(int j = 0; j < *pm; j++)
853             {
854                 printf("\nx[%d][%d]=",i,j);
855                 scanf("%d", (pa+i*(*pm)+j));
856             }
857
858 }
859 //Ham xuat ma tran su dung con tro
860 void XuatMaTran(int *pa, int *pn,int *pm)
861 {
862     int i,j;
863     for(i= 0; i < *pn; i++)
864     {
865         printf("\n");
866         for(j = 0; j < *pm; j++)
867             {
868                 printf("%3d",*(pa+i*(*pm)+j)) ;
869             }
870     }
871 }
872 //Ham cong hai ma tran voi nhau
873 void CongMaTran(int *pa,int *pb, int *pc, int *pn,int *pm)
874 {
875     int i,j;
876     for(i =0 ;i<*pn;i++)
877         for(j=0;j<*pm;j++)
878             {
879                 *(pc+i * (*pm) + j) = *(pa+i * (*pm) + j) +(*(pb+i * (*pm) + j));
880             }
881 }
882
883

```

```

1  /*Bai 1:
2  Chuong trinh nhap vao 1 chuoi ky tu.
3  ->In ra ma ascii cua tung ky tu trong chuoi
4  */
5  #include <stdio.h>
6  #include <conio.h>
7  #include <string.h>
8  #include <stdlib.h>
9
10 void main()
11 {
12     //Khoi tao vung nho 100 byte de chua chuoi ki tu
13     char *s = (char*) malloc(sizeof(char)*100);
14     int i;
15     int l;    //do dai cua chuoi ky tu
16     clrscr();
17     printf("\nNhap chuoi ky tu:");
18     gets(s);
19     l = strlen(s); //Ham lay do dai cua chuoi
20     for(i = 0; i < l; i++)
21         printf("\nMa Ascii cua ky tu %c la: %d",*(s+i),*(s+i));
22     getch();
23 }
24 /*****
25 *****/
26 /*Bai 2:
27 Chuong trinh nhap vao 1 chuoi ky tu tu ban phim
28 -> Xuat ra chuoi dao nguoc cua chuoi ky tu tren
29 */
30 #include <stdio.h>
31 #include <conio.h>
32 #include <string.h>
33 #include <stdlib.h>
34
35 void main()
36 {
37     //Khoi tao vung nho 100 byte de chua chuoi ky tu
38     char *s = (char*) malloc(sizeof(char)*100);
39     int i;
40     int l;    //do dai cua chuoi ky tu
41     clrscr();
42     printf("\nNhap chuoi ky tu:");
43     gets(s);
44     l = strlen(s); //Ham lay do dai cua chuoi
45     printf("\nChuoi dao nguoc la:");
46     for(i = l-1; i >= 0; i--) //di tu cuoi chuoi ->in nguoc lai phia truoc
47         printf("%c",*(s+i));
48     getch();
49 }
50 /*****
51 *****/
52 /*Bai 3:
53 Chuong trinh nhap vao mot chuoi ky tu.
54 ->Kiem tra chuoi do co doi xung hay khong.
55 */
56 #include <stdio.h>

```

```

57 #include <conio.h>
58 #include <string.h>
59 #include <stdlib.h>
60 //Ham kiem tra chuoi doi xung, dau vao la chuoi s
61 //Tra ve 0: neu ko doi xung nguoc lai tra ve 1
62 int KiemTraDoiXung(char *s)
63 {
64     int i;
65     int l = strlen(s);
66     for(i = 0; i < l/2; i++) //Chay tu dau den giua mang
67         if(*(s+i) != *(s+l-1-i))//So sanh phan tu dau tien voi pt cuoi cung
68             return 0; //mang khong doi xung
69     return 1; //mang doi xung
70 }
71 void main()
72 {
73     //Khoi tao vung nho 100 byte de chua chuoi ky tu
74     char *s = (char*) malloc(sizeof(char)*100);
75     int i;
76     int l; //do dai cua chuoi ky tu
77     clrscr();
78     printf("\nNhap chuoi ky tu:");
79     gets(s);
80     if(KiemTraDoiXung(s) == 0)
81         printf("\nMang khong doi xung !");
82     else
83         printf("\nMang doi xung !");
84     getch();
85 }
86 /*****
87 *****/
88 /*Bai 4:
89 Chuong trinh nhap vao mot chuoi ky tu.
90 ->Dem so lan xuat hien cua moi loai ky tu trong chuoi.
91 */
92 #include <stdio.h>
93 #include <conio.h>
94 #include <string.h>
95 #include <stdlib.h>
96 //Ham dem so lan xuat hien cua 1 ky tu trong chuoi
97 //Ket qua tra ve la so lan xuat hien
98 int DemSoLanXH(char *s, char c) //c la ky tu can kiem tra
99 {
100     int i;
101     int l = strlen(s); //do dai cua chuoi ky tu
102     int count = 0; //So lan xuat hien cua ky tu trong chuoi
103     for(i = 0; i < l; i++)
104         if(c == *(s+i))
105             count++;
106     return count;
107 }
108 //Ham main
109 void main()
110 {
111     //Khoi tao vung nho 100 byte de chua chuoi ky tu
112     char *s = (char*) malloc(sizeof(char)*100);

```

```

113     int i;
114     int l;    //do dai cua chuoi ky tu
115     clrscr();
116     printf("\nNhap chuoi ky tu:");
117     gets(s);
118     l = strlen(s);
119     if(l == 0)//chuoi rong
120         printf("\nChuoi rong !");
121     else
122         for(i = 0; i < l; i++)
123         {
124             printf("\nSo lan xuat hien cua ky tu %c trong chuoi la: %d",*(s+i), DemSoLanXH(
125                 s,*(s+i)));
126         }
127     getch();
128 }
129 /*****
130 *****/
131 /*Bai 5:
132 Chuong trinh nhap vao 1 chuoi ky tu
133 -> In ra man hinh tu ben trai nhat va phan con lai cua chuoi
134 -> In ra man hinh tu ben phai nhat va phan con lai cua chuoi
135 */
136
137 #include <stdio.h>
138 #include <conio.h>
139 #include <string.h>
140 #include <stdlib.h>
141 //Khai bao nguyen mau ham
142 void XuatTraiNhat(char *s, int x);
143 void XuatPhaiNhat(char *s, int x);
144
145 //Ham main
146 void main()
147 {
148     //Khoi tao vung nho 100 byte de chua chuoi ky tu
149     char *s = (char*) malloc(sizeof(char)*100);
150     int i,j;
151     int x,y;
152     int l;    //do dai cua chuoi ky tu
153     clrscr();
154     printf("\nNhap chuoi ky tu:");
155     gets(s);
156     l = strlen(s);
157     for(i = 0; i < l; i++)
158         if(*(s+i) == ' ')
159         {
160             x = i;
161             break;
162         }
163     for(j = l-1; j >= 0; j--)
164         if(*(s+j) == ' ')
165         {
166             y = j;
167             break;

```



```

168     }
169     XuatTraiNhat(s,x);
170     XuatPhaiNhat(s,y);
171     getch();
172 }
173
174 void XuatTraiNhat(char *s, int x)
175 {
176     int l = strlen(s);
177     int i;
178     printf("\n\Phan trai nhat:");
179     for(i = 0; i < x; i++)
180         printf("%c",*(s+i));
181
182     printf("\nPhan con lai:");
183     for(i = x+1; i < l; i++)
184         printf("%c",*(s+i));
185 }
186
187 void XuatPhaiNhat(char *s, int y)
188 {
189     int l = strlen(s);
190     int i;
191     printf("\n\Phan phai nhat:");
192     for(i = y+1; i < l; i++)
193         printf("%c",*(s+i));
194
195     printf("\nPhan con lai:");
196     for(i = 0; i < y; i++)
197         printf("%c",*(s+i));
198 }
199 /*****
200 *****/
201 /*Bai 6:
202 Chuong trinh nhap vao 1 chuoi ky tu.
203 ->In moi tu 1 dong
204 */
205 #include <stdio.h>
206 #include <conio.h>
207 #include <string.h>
208 #include <stdlib.h>
209
210 void main()
211 {
212     //Khoi tao vung nho 100 byte de chua chuoi ky tu
213     char *s = (char*) malloc(sizeof(char)*100);
214     int i;
215     int l;    //do dai cua chuoi ky tu
216     clrscr();
217     printf("\nNhap chuoi ky tu:");
218     gets(s);
219     l = strlen(s); //Ham lay do dai cua chuoi
220     printf("\nKet qua:\n");
221     for(i = 0; i < l; i++)    //di tu cuoi chuoi ->in nguoc lai phia truoc
222         if(*(s+i) == ' ')
223             printf("\n");

```

```

224         else
225             printf("%c",*(s+i));
226     getch();
227 }
228 /*****
229 *****/
230 /*Bai 10:
231 Chuong trinh nhap vao ho va ten cua 1 nguoi
232 ->Cat bo cac khoang trang khong can thiet
233 ->In ra ten cua nguoi do
234 */
235 #include <stdio.h>
236 #include <conio.h>
237 #include <string.h>
238 #include <stdlib.h>
239 //Ham xoa khoang trang tu vi tri x
240 int XoaKhoangTrangX(char *s, int x)
241 {
242     int i;
243     for(i = x; i < strlen(s)-1; i++)
244     {
245         *(s+i) = *(s+i+1);
246     }
247 }
248 }
249 //Ham xoa bo khoang trang trong chuoi
250 int XoaKhoangTrang(char *s)
251 {
252     int i,j;
253     int x=strlen(s); //vi tri khoang trang hien tai
254     //truong hop chuoi khong co du lieu
255     if(strlen(s) == 0)
256         return 0;
257     for(i = 0; i < strlen(s); i++)
258     {
259         //truong hop co khoang trang o dau chuoi
260         if(*(s+0) == ' ')
261         {
262             x = 0; //vi tri dau tien trong chuoi->xoa di
263             XoaKhoangTrangX(s,x);
264             *(s+strlen(s)-1) = NULL; //Xoa phan tu cuoi cung cua mang
265             i--;
266         }
267         else //khoang trang giua nhung tu
268             if(*(s+i) == ' ' && *(s+i+1) == ' ') //co khoang trang can xoa
269             {
270                 x = i+1;
271                 XoaKhoangTrangX(s,x);
272                 *(s+strlen(s)-1) = NULL; //Xoa noi dung tai o nho
273                 i--;
274             }
275     }
276 }
277 void main()
278 {
279     //Khoi tao vung nho 100 byte de chua chuoi ky tu

```

```
280     char *s = (char*) malloc(sizeof(char)*100);
281     int i;
282     int l;    //do dai cua chuoi ky tu
283     clrscr();
284     printf("\nNhap ho va ten:");
285     gets(s);
286     l = strlen(s); //Ham lay do dai cua chuoi
287     printf("\nMang sau khi xoa cac khoang trang thua:");
288     XoaKhoangTrang(s);
289     puts(s);
290     getch();
291 }
292
```

```

1  /*Bai 1:
2  Chuong trinh nhap vao thong tin sinh vien
3  ->In thong tin sinh vien da nhap len man hinh
4  - Xep loai hoc luc cua sinh vien
5  */
6  #include <stdio.h>
7  #include <conio.h>
8  #include <string.h>
9
10 //Khai bao cau truc sinh vien
11 typedef struct{
12     char hoten[40];
13     float dtb;
14     char *xeploai;
15 }sinhvien;
16
17 //Ham xep loai hoc luc dua vao diem trung binh cua sv
18 //Ket qua tra ve la dia chi cua chuoi hoc luc
19 char *XepLoaiDTB(float d)
20 {
21     char *xl;
22     if(d >=8 )
23     {
24         xl = "Gioi";
25         return xl;
26     }
27     if(d >= 6.5)
28     {
29         xl = "Kha";
30         return xl;
31     }
32     if(d >= 5)
33     {
34         xl = "TB";
35         return xl;
36     }
37     xl = "Khong dat";
38     return xl;
39 }
40
41 //Ham nhap vao 1 sinh vien, truyen theo kieu tham bien
42 void NhapSV(sinhvien &sv)
43 {
44     fflush(stdin);
45     printf("\nNhap ho ten sv:");
46     gets(sv.hoten);
47     fflush(stdin);
48     printf("\nNhap diem trung binh:");
49     scanf("%f",&sv.dtb);
50     sv.xeploai = XepLoaiDTB(sv.dtb);
51 }
52 //Ham xuất thông tin sinh viên
53 void XuatSV(sinhvien sv)
54 {
55     int i,j;
56     printf("%s",sv.hoten);

```

```

57     for(i = 0; i < 30-strlen(sv.hoten); i++)
58         printf(" ");
59     printf("%.1f",sv.dtb);
60     for(j = 0; j < 17; j++)
61         printf(" ");
62     puts(sv.xeploai);
63 }
64 void main()
65 {
66     sinhvien sv[100]; //khoei tao vung nho 100 sinh vien
67     int i,n;
68     clrscr();
69     printf("\nNhap so luong sinh vien:");
70     scanf("%d",&n);
71     for(i = 0; i < n; i++)
72     {
73         printf("\nNhap thong tin sv %d:",i+1);
74         NhapSV(sv[i]);
75     }
76     printf("*****\n");
77     printf("HO TEN");
78     for(i = 0; i < 30-strlen("HO TEN"); i++)
79         printf(" ");
80     printf("DTB");
81     for(i = 0; i < 20 - strlen("DTB"); i++)
82         printf(" ");
83     printf("XEP LOAI");
84     printf("\n");
85     for(i = 0; i < n; i++)
86     {
87         XuatSV(sv[i]);
88     }
89     printf("*****\n");
90     getch();
91 }
92 /*****
93 *****/
94 /*Cau 2:
95 Chuong trinh nhap vao 2 phan so
96 ->Tinh tong, hieu, tich, thuong cua hai phan so sau khi da toi gian
97 */
98 #include<stdio.h>
99 #include<conio.h>
100 #include<math.h>
101 #include<stdlib.h>
102 #include<string.h>
103
104 typedef struct{
105     int tuso;
106     int mauso;
107 }phanso;
108
109 //Ham tin UCLN cua hai so nguyen
110 int UCLN(int a, int b)
111 {
112     //Tri tuyet doi hai so a,b

```

```

113     a=abs(a);
114     b=abs(b);
115     if(a == b)
116         return a;
117     while(a != b)
118     {
119         if(a > b)
120             a = a-b;
121         else
122             b = b - a;
123     }
124     return a;
125 }
126 //Ham nhap phan so
127 void NhapPS(phanso &ps)
128 {
129     printf("\nNhap tu so:");
130     scanf("%d",&ps.tuso);
131     printf("Nhap mau so:");
132     scanf("%d",&ps.mauso);
133 }
134
135 //Ham tinh tong cua hai phan so
136 void TongPS(phanso ps1, phanso ps2)
137 {
138     int tu = ps1.tuso * ps2.mauso + ps2.tuso*ps1.mauso;
139     int mau = ps1.mauso * ps2.mauso;
140     int uc = UCLN(tu,mau);
141     printf("\nTong cua hai phan so la: %d / %d",tu/uc, mau/uc);
142 }
143 //Ham tinh hieu cua hai phan so a,b
144 void HieuPS(phanso ps1, phanso ps2)
145 {
146     int tu = ps1.tuso * ps2.mauso - ps2.tuso*ps1.mauso;
147     int mau = ps1.mauso * ps2.mauso;
148     int uc = UCLN(tu,mau);
149     printf("\nHieu cua hai phan so la: %d / %d",tu/uc, mau/uc);
150 }
151 //Ham tinh tich cua hai phan so a,b
152 void TichPS(phanso ps1, phanso ps2)
153 {
154     int tu = ps1.tuso * ps2.tuso;
155     int mau = ps1.mauso * ps2.mauso;
156     int uc = UCLN(tu,mau);
157     printf("\nTich cua hai phan so la: %d / %d",tu/uc, mau/uc);
158 }
159 //Ham tinh thuong cua hai phan so a,b
160 void ThuongPS(phanso ps1, phanso ps2)
161 {
162     int tu = ps1.tuso * ps2.mauso;
163     int mau = ps1.mauso * ps2.tuso;
164     int uc = UCLN(tu,mau);
165     printf("\nThuong cua hai phan so la: %d / %d",tu/uc, mau/uc);
166 }
167 void main()
168 {

```

```

169     phanso ps1, ps2;    //Khai bao hai phan so
170     clrscr();
171     printf("\nNhap phan so thu nhhat:");
172     NhapPS(ps1);
173     printf("\nNhap phan so thu hai:");
174     NhapPS(ps2);
175     TongPS(ps1,ps2);
176     HieuPS(ps1,ps2);
177     TichPS(ps1,ps2);
178     ThuongPS(ps1,ps2);
179     getch();
180 }
181 /*****
182 *****/
183 /*Bai 3:
184 Chuong trinh nhap vao thong tin sinh vien su dung con tro
185 ->In thong tin sinh vien da nhap len man hinh
186 - Xep loai hoc luc cua sinh vien
187 */
188 #include <stdio.h>
189 #include <conio.h>
190 #include <string.h>
191 #include <stdlib.h>
192
193 //Khai bao cau truc sinh vien
194 typedef struct{
195     char hoten[40];
196     float dtb;
197     char *xeploai;
198 }sinhvien;
199
200 //Ham xep loai hoc luc dua vao diem trung binh cua sv
201 //Ket qua tra ve la dia chi cua chuoi hoc luc
202 char *XepLoaiDTB(float d)
203 {
204     char *xl;
205     if(d >=8 )
206     {
207         xl = "Gioi";
208         return xl;
209     }
210     if(d >= 6.5)
211     {
212         xl = "Kha";
213         return xl;
214     }
215     if(d >= 5)
216     {
217         xl = "TB";
218         return xl;
219     }
220     xl = "Khong dat";
221     return xl;
222 }
223
224 //Ham nhap vao 1 sinh vien, truyen vao dia chi cua sv

```

```

225 void NhapSV(sinhvien *sv)
226 {
227     float t;
228     fflush(stdin);
229     printf("\nNhap ho ten sv:");
230     gets(sv->hoten);
231     fflush(stdin);
232     printf("\nNhap diem trung binh:");
233     scanf("%f",&t); //nhap vao bien t truoc khi dua vao dtb
234     sv->dtb = t;
235     sv->xeploai = XepLoaiDTB(sv->dtb);
236 }
237
238 //Ham xuất thông tin sinh viên
239 void XuatSV(sinhvien *sv)
240 {
241     int i,j;
242     printf("%s",sv->hoten);
243     for(i = 0; i < 30-strlen(sv->hoten); i++)
244         printf(" ");
245     printf("%.1f",sv->dtb);
246     for(j = 0; j < 17; j++)
247         printf(" ");
248     puts(sv->xeploai);
249 }
250 void main()
251 {
252     //sinhvien sv[100];
253     sinhvien *sv;
254     sv = (sinhvien*) malloc(sizeof(sinhvien)*100); //khai báo vùng nhớ lưu 100 sv
255     int i,n;
256     clrscr();
257     printf("\nNhap so luong sinh vien:");
258     scanf("%d",&n);
259     for(i = 0; i < n; i++)
260     {
261         printf("\nNhap thông tin sv %d:",i+1);
262         NhapSV(sv+i);
263     }
264     printf("*****\n");
265     printf("HO TEN");
266     for(i = 0; i < 30-strlen("HO TEN"); i++)
267         printf(" ");
268     printf("DTB");
269     for(i = 0; i < 20 - strlen("DTB"); i++)
270         printf(" ");
271     printf("XEP LOAI");
272     printf("\n");
273     for(i = 0; i < n; i++)
274     {
275         XuatSV(sv+i);
276     }
277     printf("*****\n");
278     getch();
279 }
280

```



```

1  /*Vi du 1:
2  Chuong trinh ghi chuoai ky tu len tap tin van ban.
3  */
4  #include<stdio.h>
5  #include<conio.h>
6
7  int main()
8  {
9      FILE *f;
10     clrscr();
11     f=fopen("D:\\Baihat.txt","a");
12
13     if (f!=NULL)
14     {
15         fputs("Em oi Ha Noi pho.\n",f);
16         fputs("Ta con em, mui hoang lan; ta con em, mui hoa sua.",f);
17         fclose(f);
18     }
19     getch();
20     return 0;
21 }
22 /*****
23 *****/
24 /*Vi du 2:
25 Chuong trinh chep tap tin Baihat.txt sang tap tin Baica.txt
26 */
27 #include<stdio.h>
28 #include<conio.h>
29
30 int main()
31 {
32     FILE *f1,*f2;
33     clrscr();
34     f1=fopen("D:\\Baihat.txt","r");
35     f2=fopen("D:\\Baica.txt","w");
36     if (f1!=NULL && f2!=NULL)
37     {
38         int ch=fgetc(f1);
39         while (!feof(f1))
40         {
41             fputc(ch,f2);
42             ch=fgetc(f1);
43         }
44         fcloseall();
45     }
46     getch();
47     return 0;
48 }
49 /*****
50 *****/
51 /*Vi du 3:
52 Viet chuong trinh ghi len tap tin CacSo.dat 3 gia tri so (thuc,nguyen,nguyen dai).
53 ->Sau do doc cac so tu tap tin vua ghi va hien thi len man hinh.
54 */
55 #include<stdio.h>
56 #include<conio.h>

```

```

57
58 int main()
59 {
60     FILE *f;
61     clrscr();
62     f=fopen("D:\\CacSo.txt","wb");
63     if (f!=NULL)
64     {
65         double d=3.14;
66         int i=101;
67         long l=54321;
68         fwrite(&d,sizeof(double),1,f);
69         fwrite(&i,sizeof(int),1,f);
70         fwrite(&l,sizeof(long),1,f);
71         /* Doc tu tap tin*/
72         rewind(f);
73         fread(&d,sizeof(double),1,f);
74         fread(&i,sizeof(int),1,f);
75         fread(&l,sizeof(long),1,f);
76         printf("Cac ket qua la: %f %d %ld",d,i,l);
77         fclose(f);
78     }
79     getch();
80     return 0;
81 }
82 /*****
83 *****/
84 /*Vi du 4:
85 Chuong trinh nhap 1 mang so nguyen sau do ghi len file
86 Doc mang tu file va hien thi len manh hinh.
87 */
88 #include <stdio.h>
89 #include <conio.h>
90
91 //Ham nhap cac phan tu vao mang
92 void nhap(int a[], int n)
93 {
94     for(int i=1; i<=n; i++)
95     {
96         printf("Nhap pt a[%d]: ",i);
97         scanf("%d", &a[i]);
98     }
99 }
100
101 //Ham ghi mang vao file
102 void ghifile(int a[],int n)
103 {
104     FILE *f;
105     f=fopen("bcdonline.text","wt");
106     fprintf(f,"%d",n);
107     for(int i=1;i<=n;i++)
108         fprintf(f,"%3d",a[i]);
109     fclose(f);
110 }
111
112 //Ham doc tu file vao mang

```

```

113 void docfile(int a[],int &n)
114 {
115     FILE *f;
116     f=fopen("bcdonline.text","rt");
117     fscanf(f,"%d",&n);
118     for(int i=1;i<=n;i++)
119         fscanf(f,"%d",&a[i]);
120     fclose(f);
121 }
122
123 //Ham xuất các phần tử của mảng lên màn hình
124 void xuất(int a[],int n)
125 {
126     printf("\nKet Qua Doc File:\n\n");
127     for(int i=1;i<=n;i++)
128         printf("%3d",a[i]);
129 }
130
131 //Ham main
132 int main()
133 {
134     int a[100],n;
135     clrscr();
136     printf("\nNhap so phan tu:");
137     scanf("%d",&n);
138     nhap(a,n);      //goi ham nhap mang;
139     ghifile(a,n);
140     docfile(a,n);
141     xuất(a,n);      //goi ham xuất mảng
142     getch();
143     return 0;
144 }
145 /*****
146 *****/
147 /*Vi dụ 5:
148 Mọi sinh viên cần quản lý ít nhất 2 thông tin: mã số sinh viên
149 và họ tên. Viết chương trình cho phép lựa chọn các chức năng:
150 -Nhập danh sách sinh viên từ bàn phím rồi ghi lên tập tin SinhVien.dat
151 -Đọc dữ liệu từ tập tin SinhVien.dat rồi hiển thị danh sách lên màn hình
152 -Tìm kiếm họ tên của một sinh viên nào đó dựa vào mã số sinh viên nhập từ bàn phím.
153 */
154 #include<stdio.h>
155 #include<conio.h>
156 #include<string.h>
157
158 typedef struct
159 {
160     char Ma[10];
161     char HoTen[40];
162 }SinhVien;
163
164 void WriteFile(char *FileName)
165 {
166     FILE *f;
167     int n,i;
168     SinhVien sv;

```

```

169     f=fopen(FileName,"ab");
170     printf("Nhap so luong sinh vien:");
171     scanf("%d",&n);
172     fflush(stdin);
173     for(i=1;i<=n;i++)
174     {
175         printf("Sinh vien thu %i\n",i);
176         printf("  - MSSV: ");gets(sv.Ma);
177         printf("  - Ho ten: ");gets(sv.HoTen);
178         fwrite(&sv,sizeof(sv),1,f);
179         fflush(stdin);
180     }
181     fclose(f);
182     printf("Bam phim bat ky de tiep tuc");
183     getch();
184 }
185
186 void ReadFile(char *FileName)
187 {
188     FILE *f;
189     SinhVien sv;
190     f=fopen(FileName,"rb");
191     printf("    MSSV |   Ho va ten\n");
192     fread(&sv,sizeof(sv),1,f);
193     while (!feof(f))
194     {
195         printf("    %s |   %s\n",sv.Ma,sv.HoTen);
196         fread(&sv,sizeof(sv),1,f);
197     }
198     fclose(f);
199     printf("Bam phim bat ky de tiep tuc!!!");
200     getch();
201 }
202 void Search(char *FileName)
203 {
204     char MSSV[10];
205     FILE *f;
206     int Found=0;
207     SinhVien sv;
208
209     fflush(stdin);
210     printf("Ma so sinh vien can tim: ");gets(MSSV);
211     f=fopen(FileName,"rb");
212     while (!feof(f) && Found==0)
213     {
214         fread(&sv,sizeof(sv),1,f);
215         if (strcmp(sv.Ma,MSSV)==0)
216             Found=1;
217     }
218     fclose(f);
219     if (Found == 1)
220         printf("Tim thay SV co ma %s. Ho ten la: %s",sv.Ma,sv.HoTen);
221     else
222         printf("Tim khong thay sinh vien co ma %s",MSSV);
223
224     printf("\nBam phim bat ky de tiep tuc!!!");

```

```
225     getch();
226 }
227
228 int main()
229 {
230     int c;
231     for (;;)
232     {
233         clrscr();
234         printf("1. Nhap DSSV\n");
235         printf("2. In DSSV\n");
236         printf("3. Tim kiem\n");
237         printf("4. Thoat\n");
238         printf("Ban chon 1, 2, 3, 4: "); scanf("%d",&c);
239         if(c==1)
240             WriteFile("d:\\SinhVien.Dat");
241         else if (c==2)
242             ReadFile("d:\\SinhVien.Dat");
243         else if (c==3)
244             Search("d:\\SinhVien.Dat");
245         else break;
246     }
247     return 0;
248 }
249
250
251
252
253
```