

NHÓM BÀI TẬP SỐ 5

1. Viết chương trình nhập vào một số ở hệ 16, đổi số này sang hệ 10.
2. Cho số nguyên n nhập từ bàn phím. Viết chương trình đổi số này sang các hệ nhị phân, bát phân và thập lục phân. In các kết quả ra màn hình.
3. Viết chương trình nhập vào một mảng các ký tự (chuỗi), xuất ra chuỗi đảo ngược.
4. Viết chương trình đếm số từ trong một chuỗi được nhập từ bàn phím.
5. Viết chương trình xóa bỏ khoảng trắng thừa trong một chuỗi.
6. Viết chương trình đảo ngược các từ trong chuỗi. Ví dụ chuỗi ban đầu là “hoc sinh” thì chuỗi kết quả là “sinh hoc”.
7. Viết chương trình tính tổng các ký tự số có trong chuỗi
8. Viết các hàm trích ra n ký tự đầu tiên/cuối cùng/bắt đầu tại vị trí pos .
9. Viết chương trình kiểm tra một chuỗi có phải là chuỗi con của chuỗi kia hay không?
10. Viết chương trình đổi chữ cái đầu tiên của mỗi từ trong một chuỗi thành chữ hoa.
11. Chuyển các bài tập về mảng số (nguyên, thực) sang cấp phát động.
12. Viết chương trình tính tổng các phần tử trong một mảng 2 chiều có kích thước $M \times N$.

Cho ma trận vuông $A[20][20]$, có kích thước N . Viết chương trình cho các bài 13 đến 15:

13. Viết chương trình tính tổng hai ma trận
14. Tính tổng trên các dòng, các cột, trên đường chéo chính, đường chéo phụ.
15. Sắp xếp ma trận sao cho các phần tử trên đường chéo chính tăng dần.
16. Tìm và in ra tất cả các vị trí xuất hiện của số lớn nhất trong ma trận.
17. Tìm vị trí (tọa độ dòng, cột) của số nguyên tố đầu tiên trong mảng 2 chiều m dòng, n cột.
18. Đếm số lượng số chẵn trên biên của ma trận các số thực kích thước $m \times n$.
19. Đếm số lượng giá trị “hoàng hậu” trên ma trận. Một phần tử gọi là hoàng hậu khi nó lớn nhất trên dòng, cột và 2 đường chéo.
20. Đếm số lượng điểm “yên ngựa” trên ma trận. Một phần tử gọi là “yên ngựa” khi nó lớn nhất trên dòng và nhỏ nhất trên cột.
21. Kiểm tra dòng thứ k của ma trận có giảm dần không?

22. Cho ma trận vuông $A[20][20]$, có kích thước N. Viết chương trình sắp xếp lại ma trận tăng dần theo dòng và cột, tức là:

$$A[0][0] < A[0][1] < \dots < A[0][N-1] < A[1][0] < \dots < A[N-1][N-1]$$

Ví dụ:

$$N = 3$$

$$\begin{array}{ccc} 1 & 5 & 6 \\ 4 & 9 & 7 \\ 3 & 8 & 2 \end{array} \Rightarrow \begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{array}$$

23. Viết chương trình nhập vào một số nguyên (unsigned long). Xuất ra màn hình cách đọc của số đó. Ví dụ: số 107 đọc là “Một trăm lẻ bảy”, số 1065 đọc là “Một ngàn không trăm sáu mươi lăm”.

NOTE: Fgets tính '\0'

Gets Độ dài chuỗi s không tính ký tự \0

```
1//Chuyen doi 16 sang 10
#include<stdio.h>
#include<string.h>
#include<math.h>
void Xuat(int kq);
void Nhap(char s[]);
int Xuly(char s[]);

int main()
{
    char s[100];
    Nhap(s);
    int kq = Xuly(s);
    Xuat(kq);
}

void Nhap(char s[])
{
    fgets(s, 100, stdin);
}

int Xuly(char s[])
{
    int d = 0, h;
    for (int i = 0; i < (strlen(s) - 1); i++)
    {
        if (s[i] >= 'A' && s[i] <= 'Z')
            h = s[i] - 55;
        else if (s[i] >= 'a' && s[i] <= 'z')
            h = s[i] - 32 - 55;
        else
            h = s[i] - 48;
        int u = strlen(s) - i - 2;
```

```
d = d + h * pow(16, u);
}
return d;
}
```

```
void Xuat(int kq)
{
printf("%d ", kq);
}
```

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

```
void Nhap(int& n);
void Chuyendoi(int n, char s[], char x[], int b);
```

```
int main()
{
int n;
char x[] = {'0','1','2','3','4','5','6','7','8','9','A','B','C','D',
'E','F'};
char s[100];
Nhap(n);
Chuyendoi(n, s, x, 2);
Chuyendoi(n, s, x, 8);
Chuyendoi(n, s, x, 16);
}

void Nhap(int& n)
{
scanf_s("%d", &n);
}

void Chuyendoi(int n, char s[], char x[], int b) //b se lan luot 2 8 16
{

if (n == 0)
printf("%d\n", n);
else
{
int i = 0;
while (n>0)
{
s[i] = x[n%b];
n /= b;
i++;
}
s[i] = '\0';
_strrev(s);
puts(s);
}
}
```

```
3
#include<stdio.h>
#include<string.h>
void Nhap(char s[]);
void Xuat2(char s[]);
```

```

void Xuat1(char s[]);

int main()
{
    char s[100];
    Nhap(s);
    Xuat1(s);
    Xuat2(s);
}

void Nhap(char s[])
{
    fgets(s, 100, stdin);
}

void Xuat1(char s[]) // Cach1
{
    _strrev(s);
    puts(s);
}

void Xuat2(char s[]) // Cach2
{
    _strrev(s);
    for (int i = strlen(s) - 1; i >= 0; i--)
        printf("%c", s[i]);
}

```

```

4
#include<stdio.h>
#include<string.h>
void Xuat(int kq);
void Nhap(char s[]);
int Sotu(char s[]);

int main()
{
    char s[100];
    Nhap(s);
    int kq = Sotu(s);
    Xuat(kq);
}

void Nhap(char s[])
{
    gets(s);
}

int Sotu(char s[])
{
    int dem;
    if(s[0]!=' ')// chu y neu nhap vao ky tu dau tien la khoang trang va ko la
    khoan trang
    dem = 1;
}

```

```

else
dem = 0;
for (int i = 0; i < strlen(s); i++)
if (s[i] == ' ' && ((s[i + 1] >= 'a' && s[i + 1] <= 'z') || (s[i + 1] >= 'A'
&& s[i + 1] <= 'Z'))))
//if (s[i] == ' ' && (s[i+1]!=' '))
dem++;
return dem;
}

```

```

void Xuat(int kq)
{
printf("%d", kq);
}

```

```

5#include<stdio.h>
#include<string.h>
void Nhap(char s[]);
void Xuly(char s[]);
void Xoa(char s[], int vt);
void Xuat(char s[]);

int main()
{
char s[100];
Nhap(s);
Xuly(s);
Xuat(s);
}

void Nhap(char s[])
{
gets(s);
}

void Xuly(char s[])
{
for (int i = 0; i < strlen(s); i++)//for (int i = 0; i < strlen(s) - 1;
i++)DA
if (s[i] == ' ')
{
Xoa(s, i);
i--;// vua cong vua tru de cho i giu nguyen vitri i
}
}

void Xoa(char s[], int vt)
{
int n = strlen(s) - 1;// khong tru 1
for (int i = vt; i < n; i++)//for (int i = vt; i < n-1; i++)
{
s[i] = s[i + 1];
}
s[n] = '\0';//\0 ket thuc chuoi khong can n-- nhu mang
}

void Xuat(char s[])

```

```

{
puts(s);
}
C2
#include<stdio.h>
#include<string.h>
void Nhap(char s[]);
void Xuly(char s[], char b[]);

void Xuat(char b[]);

int main()
{
char s[100];
char b[100];
Nhap(s);
Xuly(s,b);
Xuat(b);
}

void Nhap(char s[])
{
gets(s);
}

void Xuly(char s[], char b[])
{
int j=0;
for (int i = 0; i < strlen(s); i++)//for (int i = 0; i < strlen(s) - 1; i++)DA
if (s[i] != ' ')
{
b[j++]=s[i];
}
}

void Xuat(char b[])
{
puts(b);
}

```

```

6
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main()
{
int index = 0;
int i;
char a[100];
char **b = (char **)malloc(100*sizeof(char));

gets(a);

char *p;

```

```

p = strtok(a, " "); //cat chuỗi bằng các ký tự , . và space
while(p != NULL)
{
    b[index] = p;
    index++;
    p = strtok(NULL, " "); //cat chuỗi từ vị trí dừng lại trước đó
}
for (i = index-1; i>=0; i--) //in ra các từ theo thứ tự đảo ngược
printf ("%s ", b[i]);

return 0;
}

```

```

7#include<stdio.h>
#include<string.h>
void nhap(char s[]);
int tongkytuso(char s[]);
void xuat(int kq);
int main()
{
    char s[100];
    nhap(s);
    int kq=tongkytuso(s);
    xuat(kq);

}
void nhap(char s[])
{
    gets(s);
}
int tongkytuso(char s[])
{
    int t=0;
    for(int i=0; i<(strlen(s)); i++)
    {
        if(s[i]>= '0'&& s[i]<= '9' )
            t+=(s[i]-48);
    }
    return t;
}
void xuat(int kq)
{
    printf("%d",kq);
}

```

```

8
#include <stdio.h>
#include <conio.h>
#include <string.h> //<= Phải có thư viện string.h

//Prototype
void nhapChuoi(char a[]);
void xuatChuoi(char a[]);
void Xuly(char a[]); //Trích xuất ký tự!

int main()
{
    char a[100];

```

```

nhapChuoi(a);
Xuly(a);
}

void nhapChuoi(char a[])
{
printf("Nhap chuoi: ");
gets(a);
}

void xuatChuoi(char a[])
{
puts(a);
}

void Xuly(char a[])
{
char kiTuDau[100], kiTuCuoi[100], kiTuPos[100];
int n;//So luong ki tu muon trich xuat!
int h = 0, j = 0, k = 0, pos;//Tang size cho chuoi!
//

printf("\n\n\tTRICH XUAT KI TU DAU/CUOI/VI TRI\n");
//Nhap vao so ki tu muon trich xuat!
do
{
printf("Nhap vao so n ki tu muon trich: ");
scanf("%d", &n);
} while (n > strlen(a));//So luong ki tu muon trich xuat phai nho hon do dai
chuoi a!
//Trich xuat ki tu dau!
{
for (int i = 0; i < n; i++)
{
kiTuDau[h++] = a[i];
}

kiTuDau[h] = '\0';
}
//Trich xuat ki tu cuoi!
{
for (int i = strlen(a) - 1; i > strlen(a)-n-1; i--)
//for (int i = strlen(a) - 1; i >= strlen(a)-n; i--)
{
kiTuCuoi[j++] = a[i];
}

kiTuCuoi[j] = '\0';
strrev(kiTuCuoi);
}
xuatChuoi(kiTuDau);
xuatChuoi(kiTuCuoi);
//Trich xuat ki tu tai vi tri da yeu cau!
{
do

```

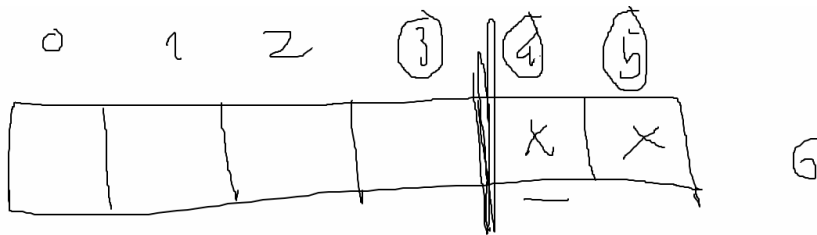


```

{
printf("\n\tChon vi tri muon trich xuat trong chuoi: pos = ");
scanf("%d", &pos); //pos vi tri trich xuat!
printf("\n\tNhap vao so n ki tu muon trich: ");
scanf("%d", &n);
} while (pos < 0 || pos > strlen(a));
for (int i = pos; i < pos + n; i++)
{
kiTuPos[k++] = a[i];
}

kiTuPos[k] = '\0';
}
xuatChuoi(kiTuPos);
}

```



```

9
#include<stdio.h>
#include<string.h>
void Xuat(int kq);
void Nhap(char S[], char s[]);
int Xuly(char S[], char s[]);

int main()
{
char S[100], s[100];
Nhap(S, s);
int kq = Xuly(S, s);
Xuat(kq);
}

void Nhap(char S[], char s[])
{
gets(S);
gets(s);
}

int Xuly(char S[], char s[])
{
if (strstr(S, s) != '\0') //if (strstr(S, s) != null)

return 1;
else
return 0;
}

void Xuat(int kq)
{
printf("%d", kq);
}

```

```
}
```

```
10
```

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

```
#include<string.h>
```

```
void Nhap(char s[]);
```

```
void doichucaidau(char s[]);
```

```
int main()
```

```
{
```

```
char s[100];
```

```
Nhap(s);
```

```
doichucaidau(s);
```

```
}
```

```
void Nhap(char s[])
```

```
{
```

```
gets(s);
```

```
}
```

```
void doichucaidau(char s[])
```

```
{
```

```
int dem;
```

```
if((s[0] >= 'a' && s[0] <= 'z'))// chu y neu nhap vao ky tu dau tien la  
khoang trang va ko la khoan trang
```

```
s[0]=s[0]-32;
```

```
for (int i = 0; i < strlen(s); i++)
```

```
if (s[i] == ' ' && (s[i+1]!=' '))
```

```
{
```

```
if((s[i+1] >= 'a' && s[i+1] <= 'z'))
```

```
{
```

```
s[i+1]=s[i+1]-32;
```

```
i++;
```

```
}
```

```
}
```

```
printf("%s",s); //phai dung printf ko dc dung puts vi puts 0d
```

```
//phai dung gets ko dung fgets vi puts 0d
```

```
}
```

```
11
```

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

```
#define MAX 500
```

```
void nhap (int &m, int &n, int A[][MAX]);
```

```
int tongpttrongmang(int m, int n, int A[][MAX]);
```

```
void xuat(int kq);
```

```
int main()
```

```
{
```

```
int m,n;
```

```
int A[MAX][MAX];
```

```
nhap(m,n,A);
```

```
int kq=tongpttrongmang(m,n,A);
```

```
xuat(kq);
```

```
}
```

```

void nhap (int &m, int &n, int A[][MAX])
{
scanf("%d%d",&m, &n);

for(int i=0;i<m;i++)
{
for(int j=0; j<n ;j++)
{
scanf("%d",&A[i][j]);
}
}
}
int tongpttrongmang(int m, int n, int A[][MAX])
{
int t=0;
for(int i=0;i<m;i++)
{
for(int j=0; j<n ;j++)
{
t+=A[i][j];
}
}
return t;
}
void xuat(int kq)
{
printf("%d",kq);
}

```

```

13#include<stdio.h>
#define MAX 20

void nhap (int &m, int A[][MAX], int B[][MAX]);
void tonghaimatran(int m, int A[][MAX], int B[][MAX]);
void xuat(int t);
void xuatdohoa();
int main()
{
int m;
int A[MAX][MAX];
int B[MAX][MAX];
nhap(m,A,B);
tonghaimatran(m,A,B);

}
void nhap (int &m, int A[][MAX], int B[][MAX])
{
scanf("%d",&m);

for(int i=0;i<m;i++)
{
for(int j=0; j<m ;j++)
{
scanf("%d",&A[i][j]);

```

```

}
}
for(int k=0;k<m;k++)
{
for(int f=0; f<m ;f++)
{
scanf("%d",&B[k][f]);
}
}
}
void tonghaimatran(int m, int A[][MAX], int B[][MAX])
{

for(int i=0;i<m;i++)
{
for(int j=0; j<m ;j++)
{
int t=(A[i][j]+B[i][j]);

xuat (t);
}

xuatohoa();
}

}
void xuat(int t)
{
printf("%d ",t);
}
void xuatohoa()
{
printf("\n");
}
}

```

```

14#include<stdio.h>
#define MAX 20
void nhap (int &m, int A[][MAX]);
void tongtrendong(int m, int A[][MAX]);
void tongtrencot(int m, int A[][MAX]);
void tongtrenduongcheochinh(int m,int A[][MAX]);
void tongtrenduongcheophu(int m, int A[][MAX]);
void xuat(int b);
int main()
{
int m;
int A[MAX][MAX];

nhap(m,A);
tongtrendong(m,A);
tongtrencot(m,A);

```

```

tongtrenduongcheochinh(m,A);
tongtrenduongcheophu(m,A);

}
void nhap (int &m, int A[][MAX])
{
scanf("%d",&m);

for(int i=0;i<m;i++)
{
for(int j=0; j<m ;j++)
{
scanf("%d",&A[i][j]);
}
}
}
void tongtrendong(int m, int A[][MAX])
{
for(int i=0;i<m;i++)
{
int sum=0;
for(int j=0; j<m ;j++)
{
sum+=A[i][j];
}
xuat(sum);
}
}
void tongtrencot(int m, int A[][MAX])
{
for(int j=0;j<m;j++)
{
int sum=0;
for(int i=0; i<m ;i++)
{
sum+=A[i][j];
}
xuat(sum);
}
}
void tongtrenduongcheochinh(int m,int A[][MAX])
{
int t=0;
for(int i=0; i<m; i++)
{
t+=A[i][i];
}
xuat(t);
}
void tongtrenduongcheophu(int m, int A[][MAX])
{
int s=0;
for(int i=0; i<m; i++)
{

```

```

s+=A[i][m-(i+1)];
}
xuat(s);
}
void xuat(int b)
{
printf("%d ",b);
}

```

```

15
#include<stdio.h>
void nhap(int& n, int a[][100]);
void xuly(int n, int a[][100]);
void hoandoi(int &a, int &b);
void xuat(int a[][100], int n);
int main()
{
int n, a[100][100];
nhap(n, a);
xuly(n, a);
xuat(a,n);
}
void nhap(int& n, int a[][100])
{
scanf_s("%d", &n);
for (int i = 0; i < n; i++)
{
for (int j = 0; j < n; j++)
{
scanf_s("%d", &a[i][j]);
}
}
}
void xuly(int n, int a[][100])
{
for (int i = 0; i < n - 1; i++)
{
for (int j = i + 1; j < n; j++)
{
if (a[i][i] > a[j][j])
hoandoi(a[i][i], a[j][j]);
}
}
}

}
void hoandoi(int& a, int& b)//& thay doi gia tri
{
int d;
d = a;
a = b;
b = d;
}
void xuat(int a[][100],int n)
{
for (int k = 0; k < n; k++)
{

```

```

for (int f = 0; f < n; f++)
{
printf("%d ", a[k][f]);
}printf("\n");
}

}

```

```

16
#include<stdio.h>
#define SIZE 20
void Nhap(int A[SIZE][SIZE], int& m, int& n);
int Max(int A[SIZE][SIZE], int m, int n);
void Xuly(int A[SIZE][SIZE], int m, int n, int kq);
void Xuat(int i, int j);

int main()
{
int A[SIZE][SIZE], m, n;
Nhap(A, m, n);
int kq = Max(A, m, n);
Xuly(A, m, n, kq);
}

void Nhap(int A[SIZE][SIZE], int& m, int& n)
{
scanf("%d%d", &m, &n);
for (int i = 0; i < m; i++)
for (int j = 0; j < n; j++)
scanf("%d", &A[i][j]);
}

int Max(int A[SIZE][SIZE], int m, int n)
{
int M = A[0][0];
for (int i = 0; i < m; i++)
for (int j = 0; j < n; j++)
if (A[i][j] > M)
M = A[i][j];
return M;
}

void Xuly(int A[SIZE][SIZE], int m, int n, int kq)
{
for (int i = 0; i < m; i++)
for (int j = 0; j < n; j++)
if (A[i][j] == kq)
Xuat(i, j);
}

void Xuat(int i, int j)
{
printf("%d %d\n", i, j);
}

```

```

17
#include<stdio.h>
void nhap(int &m, int &n, int a[][20]);
int nguyento( int x );
int songuyentodautien(int m, int n, int a[][20]);
void Xuat(int i, int j);
int main()
{
    int m,n,a[20][20];
    nhap(m,n,a);
    int kt=songuyentodautien(m,n,a);
}
void nhap(int &m, int &n, int a[][20])
{
    scanf("%d%d",&m,&n);
    for(int i=0; i<m; i++)
    {
        for(int j=0;j<n;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }
}
int nguyento( int x )
{
    int dem = 0;
    for ( int i = 1 ; i <= x ; i++)
    {
        if( x % i == 0)
            dem ++;          //      dem = dem + 1;
    }
    if (dem==2)
        return 1;
    return 0;
}

int songuyentodautien(int m, int n, int a[][20])
{
    for (int i = 0; i < m; i++)
        for (int j = 0; j < n; j++)
            if (nguyento(a[i][j]))
            {
                Xuat(i, j);
                return 0;// nguyento dau tien
            }
}
void Xuat(int i, int j)
{
    printf("%d %d\n", i, j);
}

```

```

18
#include<stdio.h>
void nhap(int &m,int&n,int a[][20]);
int sochan( int a);
int demsoluongsochan( int m, int n, int a[][20]);

```



```

void xuat(int kq);

int main()
{
    int m,n,a[20][20];
    nhap(m,n,a);
    int kq= demsoluongsochan(m,n,a);
    xuat(kq);
}

void nhap(int &m,int&n,int a[][20])
{
    scanf("%d%d", &m, &n);
    for(int i=0; i<m; i++)
        for(int j=0; j<n; j++)
            scanf("%d",&a[i][j]);
}

int sochan( int a)
{
    if(a%2==0)
        return 1;
    return 0;
}

int demsoluongsochan( int m, int n, int a[][20])
{
    int dem=0;
    for(int j=0; j<n; j++)//tren
        if(sochan(a[0][j])==1)
            dem++;
    for(int j1=0; j1<n; j1++)//duoi
        if(sochan(a[m-1][j1])==1)
            dem++;
    for(int i=0; i<m; i++)//trai
        if(sochan(a[i][0])==1)
            dem++;
    for(int i1=0; i1<m; i1++)//phai
        if(sochan(a[i1][n-1])==1)
            dem++;
    return dem;
}

void xuat(int kq)
{
    printf("%d",kq);
}

```

```

19
#include<stdio.h>
void nhap( int &m ,int &n, int a[][20]);
int soluonghoanhau(int m, int n, int a[][20]);
int lonnhattrenhang( int m ,int n, int a[][20], int gan, int i);
int lonnhattrencot( int m ,int n, int a[][20], int gan, int j);
int dcttraitren( int m ,int n, int a[][20], int gan, int x, int y);
int dcttphaidui( int m ,int n, int a[][20], int gan, int x, int y);
int dcttphaitren( int m ,int n, int a[][20], int gan, int x, int y);
int dcttraidui( int m ,int n, int a[][20], int gan, int x, int y);
void xuat (int kt);

```

```

int main()
{
int m,n,a[20][20];
nhap(m,n,a);
int kt=soluonghoanhau(m,n,a);
xuat(kt);
}
void nhap( int &m ,int &n, int a[][20])
{
scanf("%d%d", &m, &n);
for(int i=0; i<m; i++)
{
for(int j=0;j<n; j++)
{
scanf("%d",&a[i][j]);
}
}
}
int soluonghoanhau(int m, int n, int a[][20])
{
int x;
int y;
int gan;
int dem=0;
for (int i = 0; i < m; i++)
{
for (int j = 0; j < n; j++)
{
gan=a[i][j];
x=i;
y=j;
if(lonnhattrenhang(m,n,a,gan,x)==1 && lonnhattrencot(m,n,a,gan,y)==1 &&
dcttraitren(m,n,a,gan,x,y)==1 && dcttphaiduoi(m,n,a,gan,x,y)==1 &&
dcttphaitren(m,n,a,gan,x,y)==1 && dcttraidui(m,n,a,gan,x,y)==1)
dem++;
}
}
return dem;
}
int lonnhattrenhang( int m ,int n, int a[][20], int gan, int i)
{
for(int j=0; j<n; j++)
{
if(gan<=a[i][j] && a[i][j]!=gan);
return 0;
}
return 1;
}
int lonnhattrencot( int m ,int n, int a[][20], int gan, int j)
{
for(int i=0; i<m; i++)
{
if(gan<=a[i][j] && a[i][j]!=gan);
return 0;
}
}

```

```

return 1;
}
int dcttraitren( int m ,int n, int a[][20], int gan, int x, int y)
{
for(int i=x-1,j=y-1; i>=0 && j>=0; i--,j--)
{
if(a[i][j]>=gan)
return 0;
}
return 1;
}
int dcttphaidui( int m ,int n, int a[][20], int gan, int x, int y)
{
for(int i=x+1,j=y+1; i<m && j<n; i++,j++)
{
if(a[i][j]>=gan)
return 0;
}
return 1;
}
int dcttphaitren( int m ,int n, int a[][20], int gan, int x, int y)
{
for(int i=x-1,j=y+1; i>=0 && j<n; i--,j++)
{
if(a[i][j]>=gan)
return 0;
}
return 1;
}
int dcttraidui( int m ,int n, int a[][20], int gan, int x, int y)
{
for(int i=x+1,j=y-1; i<m && j>=0; i++,j--)
{
if(a[i][j]>=gan)
return 0;
}
return 1;
}
void xuat (int kt)
{
printf("%d",kt);
}
Cach 2
#include<stdio.h>
#define SIZE 5
void Nhap(int A[SIZE][SIZE], int& m, int& n);
int Soluong(int A[SIZE][SIZE], int m, int n);
void Xuat(int k);
int Phantuhoanghau(int A[SIZE][SIZE], int vitrihang, int vitricot, int
sohang, int socot);

int main()
{
int A[SIZE][SIZE], m, n;
Nhap(A, m, n);
int kq = Soluong(A, m, n);
Xuat(kq);
}

```

```

}

void Nhap(int A[SIZE][SIZE], int& m, int& n)
{
    scanf_s("%d%d", &m, &n);
    for (int i = 0; i < m; i++)
        for (int j = 0; j < n; j++)
            scanf_s("%d", &A[i][j]);
}

int Soluong(int A[SIZE][SIZE], int m, int n)
{
    int dem = 0;
    for (int i = 0; i < m; i++)
        for (int j = 0; j < n; j++)
            if (Phantuhoanghau(A, i, j, m, n))
            {
                dem++;
                printf("%d%d ", i, j);
            }
    return dem;
}

void Xuat(int k)
{
    printf("%d ", k);
}

int Phantuhoanghau(int A[SIZE][SIZE], int vitrihang, int vitricot, int
sohang, int socot)
{
    int max = A[vitrihang][vitricot];
    //Kiemtratrencot;
    for (int i = 0; i < sohang; i++)
        if (A[i][vitricot] > max)
            return 0;
    //Kiemtratrenhang;
    for (int j = 0; j < socot; j++)
        if (A[vitrihang][j] > max)
            return 0;
    //Kiemtra2duongcheo;
    ///Kiemtraduongcheotangbenphai;
    int vth = vitrihang; int vtc = vitricot;
    while (--vth >= 0 && ++vtc < socot)
        if (A[vth][vtc] > max)
            return 0;
    ///Kiemtraduongcheogiambenphai;
    vth = vitrihang; vtc = vitricot;
    while (++vth < sohang && ++vtc < socot)
    {
        if (A[vth][vtc] > max)
            return 0;
    }
    ///Kiemtraduongcheogiambentrai;
    vth = vitrihang; vtc = vitricot;
    while (++vth < sohang && --vtc >= 0)
        if (A[vth][vtc] > max)

```

```

return 0;
///Kiemtraduongcheotangbentra;
vth = vitrihang; vtc = vitricot;
while (--vth >= 0 && --vtc >= 0)
if (A[vth][vtc] > max)
return 0;
return 1;
}

```

```

20
#include<stdio.h>
#define SIZE 10
void Nhap(int A[SIZE][SIZE], int& m, int& n);
int Soluong(int A[SIZE][SIZE], int m, int n);
void Xuat(int a);
int Yenngua(int A[SIZE][SIZE], int vitrihang, int vitricot, int sohang, int
socot);

int main()
{
int A[SIZE][SIZE], m, n;
Nhap(A, m, n);
int kq = Soluong(A, m, n);
Xuat(kq);
}

void Nhap(int A[SIZE][SIZE], int& m, int& n)
{
scanf("%d%d", &m, &n);
for (int i = 0; i < m; i++)
for (int j = 0; j < n; j++)
scanf("%d", &A[i][j]);
}

int Soluong(int A[SIZE][SIZE], int m, int n)
{
int dem = 0;
for (int i = 0; i < m; i++)
for (int j = 0; j < n; j++)
if (Yenngua(A, i, j, m, n))
{
printf("%d%d ", i, j);
dem++;
}
return dem;
}

void Xuat(int a)
{
printf("%d", a);
}

int Yenngua(int A[SIZE][SIZE], int vitrihang, int vitricot, int sohang, int
socot)
{
int phantu = A[vitrihang][vitricot];
//Kiem tra tren hang__Lon nhat tren hang

```

```

for (int j = 0; j < socot; j++)
if (A[vitrihang][j] > phantu)
return 0;

//Kiem tra tren cot__Nho nhat tren cot
for (int i = 0; i < sohang; i++)
if (A[i][vitricot] < phantu)
return 0;

return 1;
}

```

```

21
#include<stdio.h>
void nhap( int &m ,int &n, int a[][20],int &k);
int kiemtratanggiam(int m,int n, int a[][20], int k);
void xuat(int kt);

int main()
{

int m,n,a[20][20],k;
nhap(m,n,a,k);
int kt=kiemtratanggiam(m,n,a,k);
xuat(kt);
}
void nhap( int &m ,int &n, int a[][20],int &k)
{
scanf("%d%d", &m, &n);
for(int i=0; i<m; i++)
{
for(int j=0;j<n; j++)
{
scanf("%d",&a[i][j]);
}
}

scanf("%d",&k);
}
int kiemtratanggiam(int m,int n, int a[][20], int k)
{
for(int j=0; j<n; j++)
{
if(a[k][j] < a[k][j+1])
return 0;
}
return 1;
}
void xuat(int kt)
{
printf("%d",kt);
}

```

22

23