

Mảng 2 chiều

	A	B	C	D	E	F	G	H
1								
2								
3	int a[3][4]			0	1	2	3	
4	int n = 3		0	1	2	3	4	
5	int m = 4		1	5	6	7	8	
6			2	9	10	11	12	
7	a[1][2] \oplus							
8								
9								

```

27 void main()
28 {
29     srand(unsigned int(time(NULL)));
30
31     int a[SIZE][SIZE];
32     int n = 0, m = 0;
33
34     do
35     {
36         printf("\nNhap so dong (n > 0): ");
37         scanf("%d", &n);
38         printf("\nNhap so cot (m > 0): ");
39         scanf("%d", &m);
40     } while (!(n > 0 && m > 0));
41
42     for (int i = 0; i < n; i++)
43     {
44         for (int j = 0; j < m; j++)
45         {
46             printf("\nNhap a[%d][%d] = ", i, j);
47             scanf("%d", &a[i][j]);
48         }
49     }
50 }
51
52

```

```
30
31     int a[SIZE][SIZE];
32     int n = 0, m = 0;
33
34     do
35     {
36         printf("\nNhap so dong (n > 0): ");
37         scanf("%d", &n);
38         printf("\nNhap so cot (m > 0): ");
39         scanf("%d", &m);
40     } while (!(n > 0 && m > 0));
41
42     for (int i = 0; i < n; i++)
43     {
44         for (int j = 0; j < m; j++)
45         {
46             printf("\nNhap a[%d][%d] = ", i, j);
47             scanf("%d", &a[i][j]);
48         }
49     }
50
51     printf("\n-----Mang 2 chieu-----\n");
52     for (int i = 0; i < n; i++)
53     {
54         for (int j = 0; j < m; j++)
55         {
56             printf("%d \t", a[i][j]);
57         }
58     }
59 }
60
```

Nhap so dong ($n > 0$): -9

Nhap so cot ($m > 0$): 6

Nhap so dong ($n > 0$): 2

Nhap so cot ($m > 0$): 3

Nhap $a[0][0] = 1$

Nhap $a[0][1] = 2$

Nhap $a[0][2] = 3$

Nhap $a[1][0] = 4$

Nhap $a[1][1] = 5$

Nhap $a[1][2] = 6$

-----Mang 2 chieu-----

1 2 3 4 5 6

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 18712) exited with code 0.

Press any key to close this window . . .

```
36     printf("\nNhap so dong (n > 0): ");
37     scanf("%d", &n);
38     printf("\nNhap so cot (m > 0): ");
39     scanf("%d", &m);
40 } while (!(n > 0 && m > 0));
41
42 for (int i = 0; i < n; i++)
43 {
44     for (int j = 0; j < m; j++)
45     {
46         printf("\nNhap a[%d][%d] = ", i, j);
47         scanf("%d", &a[i][j]);
48     }
49 }
50
51 printf("\n-----Mang 2 chieu-----\n");
52 for (int i = 0; i < n; i++)
53 {
54     for (int j = 0; j < m; j++)
55     {
56         printf("%d \t", a[i][j]);
57     }
58     printf("\n");
59 }
60 }
61
```

```
36     printf("\nNhap so dong (n > 0): ");
37     scanf("%d", &n);
38     printf("\nNhap so cot (m > 0): ");
39     scanf("%d", &m);
40 } while (!(n > 0 && m > 0));
41
42 for (int i = 0; i < n; i++)
43 {
44     for (int j = 0; j < m; j++)
45     {
46         // printf("\nNhap a[%d][%d] = ", i, j);
47         // scanf("%d", &a[i][j]);
48         a[i][j] = rand() % 20 + 1;
49     }
50 }
51
52 printf("\n-----Mang 2 chieu-----\n");
53 for (int i = 0; i < n; i++)
54 {
55     for (int j = 0; j < m; j++)
56     {
57         printf("%d\t", a[i][j]);
58     }
59     printf("\n");
60 }
61 }
```

```

1  #include <stdio.h>
2  #include <math.h>
3
4  #include <stdlib.h>
5  #include <time.h>
6
7  #define SIZE 100
8
9  int KiemTraSoNguyenTo(int n)
10 {
11     if (n == 2)
12         return 1;
13     if (n % 2 == 0 || n < 2)
14         return 0;
15
16     for (int i = 2; i < n; i++)
17     {
18         if (n % i == 0)
19         {
20             return 0;
21         }
22     }
23
24     return 1;
25 }
26
27
28 void main()
29 {
30     srand(unsigned int(time(NULL)));
31
32     int a[SIZE][SIZE];
33     int n = 0, m = 0;
34
35     do
    
```

```

43     for (int i = 0; i < n; i++)
44     {
45         for (int j = 0; j < m; j++)
46         {
47             // printf("\nNhap a[%d][%d] = ", i, j);
48             // scanf("%d", &a[i][j]);
49             a[i][j] = rand() % 20 + 1;
50         }
51     }
52
53     printf("\n-----Mang 2 chieu-----\n");
54     for (int i = 0; i < n; i++)
55     {
56         for (int j = 0; j < m; j++)
57         {
58             printf("%d \t", a[i][j]);
59         }
60         printf("\n");
61     }
62
63     printf("\n-----Phan tu nguyen to-----\n");
64     for (int i = 0; i < n; i++)
65     {
66         for (int j = 0; j < m; j++)
67         {
68             if (KiemTraSoNguyenTo(a[i][j]) == 1)
69                 printf("%d \t", a[i][j]);
70         }
71     }
72 }
73

```


Nhap so dong ($n > 0$): 3

Nhap so cot ($m > 0$): 4

-----Mang 2 chieu-----

16 5 6 11

7 11 3 6

17 5 7 12

-----Phan tu nguyen to-----

5 11 7 11 3 17 5 7

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 22716) exited with code 0.

Press any key to close this window . . .

```
55 {
56     for (int j = 0; j < m; j++)
57     {
58         printf("%d \t", a[i][j]);
59     }
60     printf("\n");
61 }
62
63 printf("\n-----Phan tu nguyen to-----\n");
64 for (int i = 0; i < n; i++)
65 {
66     for (int j = 0; j < m; j++)
67     {
68         if (KiemTraSoNguyenTo(a[i][j]) == 1)
69             printf("%d \t", a[i][j]);
70     }
71 }
72
73 // Bài tập:
74 // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
75 // 2. In những phần tử là số nguyên tố
76 // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
77 // 4. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
78 // 5. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
79 // 6. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
80 // 7. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
81 }
82
```

```

61     }
62
63     printf("\n-----Phan tu nguyen to-----\n");
64     for (int i = 0; i < n; i++)
65     {
66         for (int j = 0; j < m; j++)
67         {
68             if (KiemTraSoNguyenTo(a[i][j]) == 1)
69                 printf("%d \t", a[i][j]);
70         }
71     }
72
73     // Bài tập:
74     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
75     // 2. In những phần tử là số nguyên tố
76     // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
77     // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
78     // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
79
80     // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
81     // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
82     // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
83     // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
84 }
85

```

```
25     }
26
27     void NhapDongCot(int& n, int& m)
28     {
29         do
30         {
31             printf("\nNhap so dong (n > 0): ");
32             scanf("%d", &n);
33             printf("\nNhap so cot (m > 0): ");
34             scanf("%d", &m);
35         } while (!(n > 0 && m > 0));
36     }
37
38     void NhapMang()
39
40     void main()
41     {
42         srand(unsigned int(time(NULL)));
43
44         int a[SIZE][SIZE];
45         int n = 0, m = 0;
46
47
48
49         for (int i = 0; i < n; i++)
50         {
51             for (int j = 0; j < m; j++)
52             {
53                 // printf("\nNhap a[%d][%d] = ", i, j);
54                 // scanf("%d", &a[i][j]);
55                 a[i][j] = rand() % 20 + 1;
56             }
57         }
58
59         printf("\n-----Mang 2 chieu-----\n");
```

```

29     do
30     {
31         printf("\nNhap so dong (n > 0): ");
32         scanf("%d", &n);
33         printf("\nNhap so cot (m > 0): ");
34         scanf("%d", &m);
35     } while (!(n > 0 && m > 0));
36 }
37
38 void NhapMang(int a[SIZE][SIZE], int n, int m)
39 {
40     for (int i = 0; i < n; i++)
41     {
42         for (int j = 0; j < m; j++)
43         {
44             // printf("\nNhap a[%d][%d] = ", i, j);
45             // scanf("%d", &a[i][j]);
46             a[i][j] = rand() % 20 + 1;
47         }
48     }
49 }
50
51 void main()
52 {
53     srand(unsigned int(time(NULL)));
54
55     int a[SIZE][SIZE];
56     int n = 0, m = 0;
57
58
59
60
61
62     printf("\n-----Mang 2 chieu-----\n");
63     for (int i = 0; i < n; i++)
    
```

```
44 // printf("\nNhap a[%d][%d] = ", i, j);
45 // scanf("%d", &a[i][j]);
46 a[i][j] = rand() % 20 + 1;
47 }
48 }
49 }
50
51 void XuatMang(int a[SIZE][SIZE], int n, int m)
52 {
53     printf("\n-----Mang 2 chieu-----\n");
54     for (int i = 0; i < n; i++)
55     {
56         for (int j = 0; j < m; j++)
57         {
58             printf("%d \t", a[i][j]);
59         }
60         printf("\n");
61     }
62 }
63
64 void main()
65 {
66     srand(unsigned int(time(NULL)));
67
68     int a[SIZE][SIZE];
69     int n = 0, m = 0;
70
71
72
73
74
75
76
77     printf("\n-----Phan tu nguyen to-----\n");
78     for (int i = 0; i < n; i++)
```

```
50
51 void XuatMang(int a[SIZE][SIZE], int n, int m)
52 {
53     printf("\n-----Mang 2 chieu-----\n");
54     for (int i = 0; i < n; i++)
55     {
56         for (int j = 0; j < m; j++)
57         {
58             printf("%d \t", a[i][j]);
59         }
60         printf("\n");
61     }
62 }
63
64 void main()
65 {
66     srand(unsigned int(time(NULL)));
67
68     int a[SIZE][SIZE];
69     int n = 0, m = 0;
70
71     NhapDongCot(n, m);
72     NhapMang(a, n, m);
73     XuatMang(a, n, m);
74
75
76
77
78
79     printf("\n-----Phan tu nguyen to-----\n");
80     for (int i = 0; i < n; i++)
81     {
82         for (int j = 0; j < m; j++)
83         {
84             if (KiemTraSoNguyenTo(a[i][j]) == 1)
```

```

59     }
60     printf("\n");
61 }
62 }
63
64 void InPhanTuNguyenTo(int a[SIZE][SIZE], int n, int m)
65 {
66     printf("\n-----Phan tu nguyen to-----\n");
67     for (int i = 0; i < n; i++)
68     {
69         for (int j = 0; j < m; j++)
70         {
71             if (KiemTraSoNguyenTo(a[i][j]) == 1)
72                 printf("%d \t", a[i][j]);
73         }
74     }
75 }

```

```

76
77 void main()
78 {
79     srand(unsigned int(time(NULL)));
80
81     int a[SIZE][SIZE];
82     int n = 0, m = 0;
83
84     NhapDongCot(n, m);
85     NhapMang(a, n, m);
86     XuatMang(a, n, m);
87
88
89

```



```
65 {
66     printf("\n-----Phan tu nguyen to-----\n");
67     for (int i = 0; i < n; i++)
68     {
69         for (int j = 0; j < m; j++)
70         {
71             if (KiemTraSoNguyenTo(a[i][j]) == 1)
72                 printf("%d \t", a[i][j]);
73         }
74     }
75 }
76
77 void main()
78 {
79     srand(unsigned int(time(NULL)));
80
81     int a[SIZE][SIZE];
82     int n = 0, m = 0;
83
84     NhapDongCot(n, m);
85     NhapMang(a, n, m);
86     XuatMang(a, n, m);
87
88     InPhanTuNguyenTo(a, n, m);
89
90
91
92
93
94 // Bài tập:
95 // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
96 // 2. In những phần tử là số nguyên tố
97 // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
98 // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
99 // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
```

Nhap so dong ($n > 0$): 3

Nhap so cot ($m > 0$): 2

-----Mang 2 chieu-----

9 3

11 7

14 16

-----Phan tu nguyen to-----

3 11 7

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 25556) exited with code 0.

Press any key to close this window . . .

```

65 {
66     printf("\n-----Phan tu nguyen to-----\n");
67     for (int i = 0; i < n; i++)
68     {
69         for (int j = 0; j < m; j++)
70         {
71             if (KiemTraSoNguyenTo(a[i][j]) == 1)
72                 printf("%d \t", a[i][j]);
73         }
74     }
75 }
76

```

```

77 float TinhTong(int a[SIZE][SIZE], int n, int m)
78 {
79     int s = 0;
80     for (int i = 0; i < n; i++)
81     {
82         for (int j = 0; j < m; j++)
83         {
84             s += a[i][j];
85         }
86     }
87     return s;
88 }

```

```

89
90 void main()
91 {
92     srand(unsigned int(time(NULL)));
93
94     int a[SIZE][SIZE];
95     int n = 0, m = 0;
96
97     NhapDongCot(n, m);
98     NhapMang(a, n, m);
99     XuatMang(a, n, m);

```

```

71         if (KiemTraSoNguyenTo(a[i][j]) == 1)
72             printf("%d \t", a[i][j]);
73     }
74 }
75 }
76
77 int TinhTong(int a[SIZE][SIZE], int n, int m)
78 {
79     int s = 0;
80     for (int i = 0; i < n; i++)
81     {
82         for (int j = 0; j < m; j++)
83         {
84             s += a[i][j];
85         }
86     }
87     return s;
88 }
89
90 void main()
91 {
92     srand(unsigned int(time(NULL)));
93
94     int a[SIZE][SIZE];
95     int n = 0, m = 0;
96
97     NhapDongCot(n, m);
98     NhapMang(a, n, m);
99     XuatMang(a, n, m);
100
101     InPhanTuNguyenTo(a, n, m);
102
103     // Bài tập:
104     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
105     // 2. In những phần tử là số nguyên tố

```

```
92 srand(unsigned int(time(NULL)));
93
94 int a[SIZE][SIZE];
95 int n = 0, m = 0;
96
97 NhapDongCot(n, m);
98 NhapMang(a, n, m);
99 XuatMang(a, n, m);
100
101 InPhanTuNguyenTo(a, n, m);
102
103 // Bài tập:
104 // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
105 // 2. In những phần tử là số nguyên tố
106
107 // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
108 float TB = TinhTong(a, n, m) / (n * m * 1.0);
109 printf("\nTB cong = %.2f", TB);
110
111 // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
112 // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
113
114 // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
115 // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
116 // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
117 // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
118 }
119
```

```
92 srand(unsigned int(time(NULL)));
93
94 int a[SIZE][SIZE];
95 int n = 0, m = 0;
96
97 NhapDongCot(n, m);
98 NhapMang(a, n, m);
99 XuatMang(a, n, m);
100
101 // InPhanTuNguyenTo(a, n, m);
102
103 // Bài tập:
104 // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
105 // 2. In những phần tử là số nguyên tố
106
107 // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
108 float TB = TinhTong(a, n, m) / (n * m * 1.0);
109 printf("\nTB cong = %.2f", TB);
110
111 // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
112 // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
113
114 // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
115 // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
116 // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
117 // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
118 }
119
```

Nhap so dong (n > 0): 2

Nhap so cot (m > 0): 2

-----Mang 2 chieu-----

4 2

5 12

TB cong = 5.75

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 24580) exited with code 0.

Press any key to close this window . . .

```
89
90 int TimMax(int a[SIZE][SIZE], int n, int m)
91 {
92     int Max = a[0][0];
93     for (int i = 0; i < n; i++)
94     {
95         for (int j = 0; j < m; j++)
96         {
97             if (a[i][j] > Max)
98             {
99                 Max = a[i][j];
100             }
101         }
102     }
103     return Max;
104 }
```

*tìm giá trị lớn
nhất trong
mảng 2 chiều*

```
105
106 void main()
107 {
108     srand(unsigned int(time(NULL)));
109
110     int a[SIZE][SIZE];
111     int n = 0, m = 0;
112
113     NhapDongCot(n, m);
114     NhapMang(a, n, m);
115     XuatMang(a, n, m);
116
117     // InPhanTuNguyenTo(a, n, m);
118
119     // Bài tập:
120     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
121     // 2. In những phần tử là số nguyên tố
122
123     // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
```



```

107 {
108     srand(unsigned int(time(NULL)));
109
110     int a[SIZE][SIZE];
111     int n = 0, m = 0;
112
113     NhapDongCot(n, m);
114     NhapMang(a, n, m);
115     XuatMang(a, n, m);
116
117     // InPhanTuNguyenTo(a, n, m);
118
119     // Bài tập:
120     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
121     // 2. In những phần tử là số nguyên tố
122
123     // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
124     // float TB = TinhTong(a, n, m) / (n * m * 1.0);
125     // printf("\nTB cong = %.2f", TB);
126
127     // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
128     int Max = TimMax(a, n, m);
129     printf("\nMax = %d", Max);
130
131     // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
132
133     // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
134     // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
135     // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
136     // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
137 }
138

```

Nhap so dong ($n > 0$): 3

Nhap so cot ($m > 0$): 5

-----Mang 2 chieu-----

13	9	13	3	13
2	7	16	18	16
5	11	12	4	16

Max = 18

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 19060) exited with code 0.

Press any key to close this window . . .

```

107 {
108     srand(unsigned int(time(NULL)));
109
110     int a[SIZE][SIZE];
111     int n = 0, m = 0;
112
113     NhapDongCot(n, m);
114     NhapMang(a, n, m);
115     XuatMang(a, n, m);
116
117     // InPhanTuNguyenTo(a, n, m);
118
119     // Bài tập:
120     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
121     // 2. In những phần tử là số nguyên tố
122
123     // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
124     // float TB = TinhTong(a, n, m) / (n * m * 1.0);
125     // printf("\nTB cong = %0.2f", TB);
126
127     // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
128     /*int Max = TimMax(a, n, m);
129     printf("\nMax = %d", Max);*/
130
131     // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
132
133     // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
134
135     // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
136     // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
137     // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
138 }
139

```

[illegible]

```
95     for (int j = 0; j < m; j++)
96     {
97         if (a[i][j] > Max)
98         {
99             Max = a[i][j];
100         }
101     }
102 }
103 return Max;
104 }
105
106 void InPhanTuTrenDong(int a[SIZE][SIZE], int n, int m, int Dong)
107 {
108     printf("\n-----Phan tu tren dong: %d-----\n", Dong);
109     for (int i = 0; i < n; i++)
110     {
111         for (int j = 0; j < m; j++)
112         {
113             if (i == Dong)
114                 printf("%d \t", a[i][j]);
115         }
116     }
117 }
118 void main()
119 {
120     srand(unsigned int(time(NULL)));
121
122     int a[SIZE][SIZE];
123     int n = 0, m = 0;
124
125     NhapDongCot(n, m);
126     NhapMang(a, n, m);
127     XuatMang(a, n, m);
128
129     // InPhanTuNguyenTo(a, n, m);
```

```
122     int a[SIZE][SIZE];
123     int n = 0, m = 0;
124
125     NhapDongCot(n, m);
126     NhapMang(a, n, m);
127     XuatMang(a, n, m);
128
129     // InPhanTuNguyenTo(a, n, m);
130
131     // Bài tập:
132     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
133     // 2. In những phần tử là số nguyên tố
134
135     // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
136     // float TB = TinhTong(a, n, m) / (n * m * 1.0);
137     // printf("\nTB cong = %.2f", TB);
138
139     // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
140     /*int Max = TimMax(a, n, m);
141     printf("\nMax = %d", Max);*/
142
143     // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
144
145     // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
146     int Dong = 1;
147     InPhanTuTrenDong(a, n, m, Dong);
148     // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
149     // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
150     // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
151 }
152
```

Nhap so dong (n > 0): 3

Nhap so cot (m > 0): 5

-----Mang 2 chieu-----

13	5	14	1	2
9	16	14	18	11
4	4	16	11	16

-----Phan tu tren dong: 1-----

9	16	14	18	11
---	----	----	----	----

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 13672) exited with code 0.
Press any key to close this window . . .

```

98     {
99         Max = a[i][j];
100     }
101 }
102 }
103 return Max;
104 }
105


```

```

106 void InPhanTuTrenDong(int a[SIZE][SIZE], int n, int m, int Dong)
107 {
108     printf("\n-----Phan tu tren dong: %d-----\n", Dong);
109     for (int j = 0; j < m; j++)
110     {
111         printf("%d \t", a[Dong][j]);
112     }
113 }

```

*hoặc có thể làm theo
cách sau (gán dòng)*



```

114 void main()
115 {
116     srand(unsigned int(time(NULL)));
117
118     int a[SIZE][SIZE];
119     int n = 0, m = 0;
120
121     NhapDongCot(n, m);
122     NhapMang(a, n, m);
123     XuatMang(a, n, m);
124
125     // InPhanTuNguyenTo(a, n, m);
126
127     // Bài tập:
128     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
129     // 2. In những phần tử là số nguyên tố
130
131     // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
132     // float TB = TinhTong(a, n, m) / (n * m * 1.0);

```



```

122     NhapMang(a, n, m);
123     XuatMang(a, n, m);
124
125     // InPhanTuNguyenTo(a, n, m);
126
127     // Bài tập:
128     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
129     // 2. In những phần tử là số nguyên tố
130
131     // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
132     // float TB = TinhTong(a, n, m) / (n * m * 1.0);
133     // printf("\nTB cong = %0.2f", TB);
134
135     // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
136     /*int Max = TimMax(a, n, m);
137     printf("\nMax = %d", Max);*/
138
139     // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
140
141     // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
142     int Dong = 0;
143     InPhanTuTrenDong(a, n, m, Dong);
144     // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
145     // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
146     // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
147 }
148

```

Nhap so dong ($n > 0$): 2

Nhap so cot ($m > 0$): 3

-----Mang 2 chieu-----

12	16	14
----	----	----

5	16	17
---	----	----

-----Phan tu tren dong: 0-----

12	16	14
----	----	----

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 22000) exited with code 0.

Press any key to close this window . . .

```

104 }
105
106 void InPhanTuTrenDong(int a[SIZE][SIZE], int n, int m, int Dong)
107 {
108     printf("\n-----Phan tu tren dong: %d-----\n", Dong);
109     for (int j = 0; j < m; j++)
110     {
111         printf("%d \t", a[Dong][j]);
112     }
113 }
114
115 void InPhanTuTrenCot(int a[SIZE][SIZE], int n, int m, int Cot)
116 {
117     printf("\n-----Phan tu tren cot: %d-----\n", Cot);
118     for (int i = 0; i < n; i++)
119     {
120         printf("%d \t", a[i][Cot]);
121     }
122 }
123
124 void main()
125 {
126     srand(unsigned int(time(NULL)));
127
128     int a[SIZE][SIZE];
129     int n = 0, m = 0;
130
131     NhapDongCot(n, m);
132     NhapMang(a, n, m);
133     XuatMang(a, n, m);
134
135     // InPhanTuNguyenTo(a, n, m);
136
137     // Bài tập:
138     // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
139     // 2. In những phần tử là số nguyên tố

```

```
128 int n = 0, m = 0;
129
130 NhapDongCot(n, m);
131 NhapMang(a, n, m);
132 XuatMang(a, n, m);
133
134 // InPhanTuNguyenTo(a, n, m);
135
136 // Bài tập:
137 // 1. Viết hàm nhập, random, xuất các phần tử trong mảng 2 chiều
138 // 2. In những phần tử là số nguyên tố
139
140 // 3. Tính trung bình cộng các giá trị trong mảng 2 chiều.
141 // float TB = TinhTong(a, n, m) / (n * m * 1.0);
142 // printf("\nTB cong = %0.2f", TB);
143
144 // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
145 /*int Max = TimMax(a, n, m);
146 printf("\nMax = %d", Max);*/
147
148 // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
149
150 // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
151 /*int Dong = 0;
152 InPhanTuTrenDong(a, n, m, Dong);*/
153
154 // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
155 int Cot = 2;
156 InPhanTuTrenCot(a, n, m, Cot);
157
158 // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
159 // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
160 }
161
```

```
Select Microsoft Visual Studio Debug Console

Nhap so dong (n > 0): 2

Nhap so cot (m > 0): 4

-----Mang 2 chieu-----
7      13      11      14
16     20      3      18

-----Phan tu tren cot: 2-----
11      3

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 12736) exited with code 0.
Press any key to close this window . . .
```

```
107 {
108     printf("\n-----Phan tu tren dong: %d-----\n", Dong);
109     for (int j = 0; j < m; j++)
110     {
111         printf("%d \t", a[Dong][j]);
112     }
113 }
114
115 int MaxDong(int a[SIZE][SIZE], int n, int m, int Dong)
116 {
117     int Max = a[Dong][0];
118     for (int j = 1; j < m; j++)
119     {
120         if (Max < a[Dong][j])
121             Max = a[Dong][j];
122     }
123     return Max;
124 }
125
126 void InPhanTuTrenCot(int a[SIZE][SIZE], int n, int m, int Cot)
127 {
128     printf("\n-----Phan tu tren cot: %d-----\n", Cot);
129     for (int i = 0; i < n; i++)
130     {
131         printf("%d \t", a[i][Cot]);
132     }
133 }
134 void main()
135 {
136     srand(unsigned int(time(NULL)));
137
138     int a[SIZE][SIZE];
139     int n = 0, m = 0;
140
141     NhapDongCot(n, m);
```

```
152 // float TB = TinhTong(a, n, m) / (n * m * 1.0);
153 // printf("\nTB cong = %.2f", TB);
154
155 // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
156 /*int Max = TimMax(a, n, m);
157 printf("\nMax = %d", Max);*/
158
159 // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
160
161 // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
162 /*int Dong = 0;
163 InPhanTuTrenDong(a, n, m, Dong);*/
164
165 // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
166 /*int Cot = 2;
167 InPhanTuTrenCot(a, n, m, Cot);*/
168
169 // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
170 int MaxDong = TimMax(a, n, m);
171 printf("\nMax = %d", MaxDong);
172
173 // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
174 }
175
```

```

107 {
108     printf("\n-----Phan tu tren dong: %d-----\n", Dong);
109     for (int j = 0; j < m; j++)
110     {
111         printf("%d \t", a[Dong][j]);
112     }
113 }
114
115 int TimMaxDong(int a[SIZE][SIZE], int n, int m, int Dong)
116 {
117     int Max = a[Dong][0];
118     for (int j = 1; j < m; j++)
119     {
120         if (Max < a[Dong][j])
121             Max = a[Dong][j];
122     }
123     return Max;
124 }
125
126 void InPhanTuTrenCot(int a[SIZE][SIZE], int n, int m, int Cot)
127 {
128     printf("\n-----Phan tu tren cot: %d-----\n", Cot);
129     for (int i = 0; i < n; i++)
130     {
131         printf("%d \t", a[i][Cot]);
132     }
133 }
134 void main()
135 {
136     srand(unsigned int(time(NULL)));
137
138     int a[SIZE][SIZE];
139     int n = 0, m = 0;
140
141     NhapDongCot(n, m);

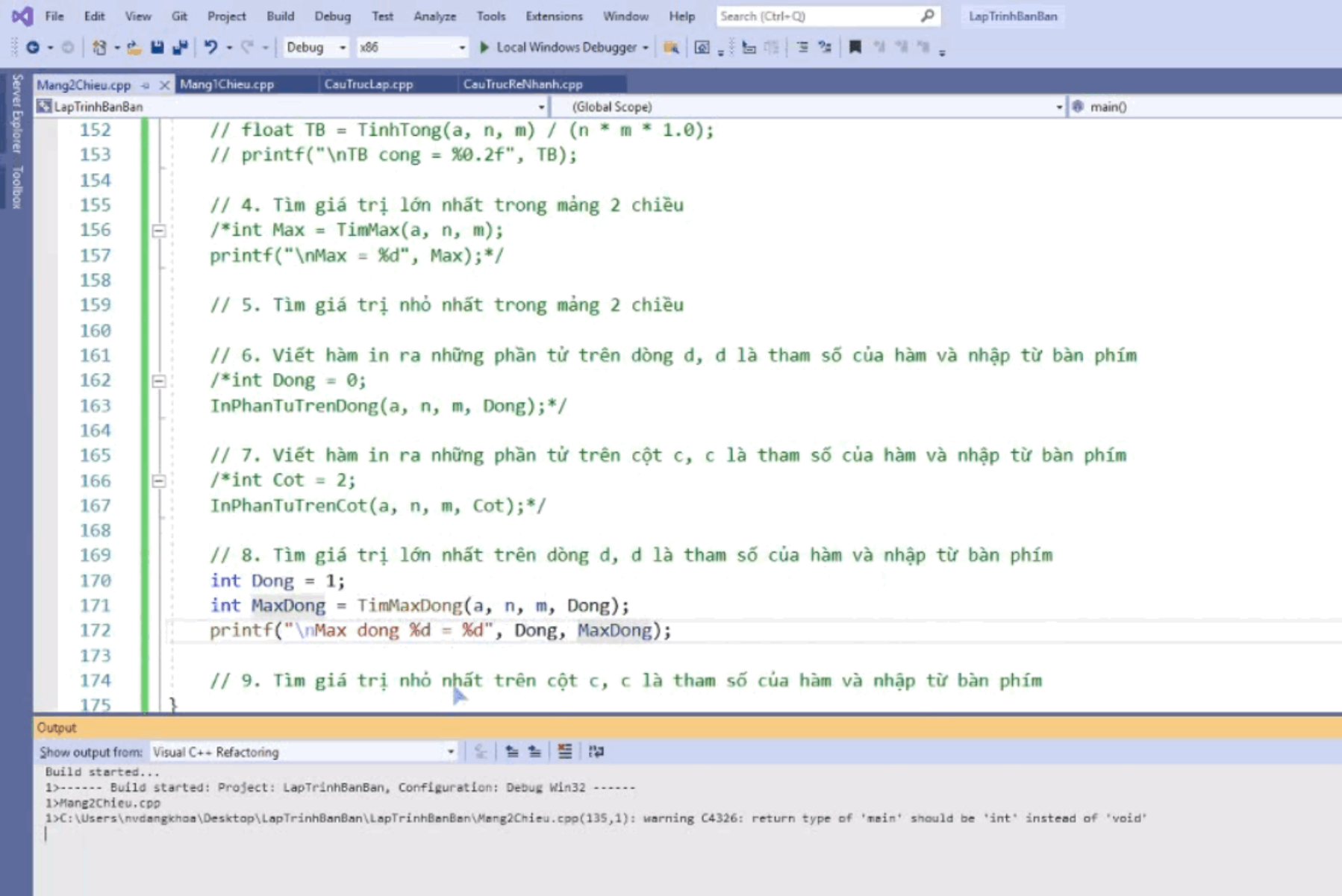
```



```
152 // float TB = TinhTong(a, n, m) / (n * m * 1.0);
153 // printf("\nTB cong = %.2f", TB);
154
155 // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
156 /*int Max = TimMax(a, n, m);
157 printf("\nMax = %d", Max);*/
158
159 // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
160
161 // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
162 /*int Dong = 0;
163 InPhanTuTrenDong(a, n, m, Dong);*/
164
165 // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
166 /*int Cot = 2;
167 InPhanTuTrenCot(a, n, m, Cot);*/
168
169 // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
170 int MaxDong = TimMaxDong(a, n, m);
171 printf("\nMax = %d", MaxDong);
172
173 // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
174 }
175
```

Mang2Chieu.cpp Mang1Chieu.cpp CauTrucLap.cpp CauTrucReNhanh.cpp LapTrinhBanBan (Global Scope) main()

```
152 // float TB = TinhTong(a, n, m) / (n * m * 1.0);
153 // printf("\nTB cong = %.2f", TB);
154
155 // 4. Tìm giá trị lớn nhất trong mảng 2 chiều
156 /*int Max = TimMax(a, n, m);
157 printf("\nMax = %d", Max);*/
158
159 // 5. Tìm giá trị nhỏ nhất trong mảng 2 chiều
160
161 // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
162 /*int Dong = 0;
163 InPhanTuTrenDong(a, n, m, Dong);*/
164
165 // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
166 /*int Cot = 2;
167 InPhanTuTrenCot(a, n, m, Cot);*/
168
169 // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
170 int Dong = 1;
171 int MaxDong = TimMaxDong(a, n, m, Dong);
172 printf("\nMax dong %d = %d", MaxDong);
173
174 // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
175 }
176
```



Nhap so dong ($n > 0$): 3

Nhap so cot ($m > 0$): 5

-----Mang 2 chieu-----

1	8	4	7	1
12	4	7	9	17
20	2	2	12	10

Max dong 1 = 17

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 25500) exited with code 0.

Press any key to close this window . . .

5.4. MA TRẬN VUÔNG

5.4.1. *Khái niệm*

Là ma trận có số dòng và số cột bằng nhau.

5.4.2. *Ma trận đơn vị*

Ma trận đơn vị I_n là ma trận vuông kích thước $(n \times n)$ có tất cả các phần tử nằm trên đường chéo chính bằng 1 và những phần tử còn lại bằng 0.

1	0	0
0	1	0
0	0	1



```
27 void NhapDongCot(int& n, int& m)
28 {
29     do
30     {
31         printf("\nNhap so dong (n > 0): ");
32         scanf("%d", &n);
33         /*printf("\nNhap so cot (m > 0): ");
34         scanf("%d", &m);*/
35         // } while (!(n > 0 && m > 0));
36     } while (!(n > 0));
37     m = n;
38 }
```

do mảng vuông thì dòng cột bằng nhau nên chỉ cần nhập n đã đủ

```
40 void NhapMang(int a[SIZE][SIZE], int n, int m)
41 {
42     for (int i = 0; i < n; i++)           không cần biến m
43     {
44         for (int j = 0; j < m; j++)
45         {
46             // printf("\nNhap a[%d][%d] = ", i, j);
47             // scanf("%d", &a[i][j]);
48             a[i][j] = rand() % 20 + 1;
49         }
50     }
51 }
```

```
52
53 void XuatMang(int a[SIZE][SIZE], int n, int m)
54 {
55     printf("\n-----Mang 2 chieu-----\n");
56     for (int i = 0; i < n; i++)           không cần biến m
57     {
58         for (int j = 0; j < m; j++)
59         {
60             printf("%d \t", a[i][j]);
61         }
62         printf("\n");
63     }
64 }
```

Nhap so dong ($n > 0$): 4

-----Mang 2 chieu-----

17	11	18	11
14	12	11	4
11	1	11	3
12	20	3	9

C:\Users\nvdangkhoa\Desktop\LapTrinhBanBan\Debug\LapTrinhBanBan.exe (process 12964) exited with code 0.
Press any key to close this window . . .

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12
3	13	14	15	16

	0	1	2	3
0	00	01	02	03
1	10	11	12	13
2	20	21	22	23
3	30	31	32	33

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12
3	13	14	15	16

	0	1	2	3
0	 00	 01	 02	 03
1	10	11	12	13
2	20	21	22	23
3	30	31	32	33

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12
3	13	14	15	16

	0	1	2	3
0	*	*	*	*
1	*	*	*	
2	*	*		
3	*			

	0	1	2	3
0	00	01	02	03
1	10	11	12	13
2	20	21	22	23
3	30	31	32	33




```
164
165 // 6. Viết hàm in ra những phần tử trên dòng d, d là tham số của hàm và nhập từ bàn phím
166 /*int Dong = 0;
167 InPhanTuTrenDong(a, n, m, Dong);*/
168
169 // 7. Viết hàm in ra những phần tử trên cột c, c là tham số của hàm và nhập từ bàn phím
170 /*int Cot = 2;
171 InPhanTuTrenCot(a, n, m, Cot);*/
172
173 // 8. Tìm giá trị lớn nhất trên dòng d, d là tham số của hàm và nhập từ bàn phím
174 /*int Dong = 1;
175 int MaxDong = TimMaxDong(a, n, m, Dong);
176 printf("\nMax dong %d = %d", Dong, MaxDong);*/
177
178 // 9. Tìm giá trị nhỏ nhất trên cột c, c là tham số của hàm và nhập từ bàn phím
179
180 // 1. Thi và sửa bài trên lớp
181 // 2. Thi ở nhà
182
183 // mảng 2 chiều
184 // chuỗi ký tự: 1 buổi
185 // cấu trúc: 2 buổi
186 //
187 }
188
```