

BÀI THỰC HÀNH 4

Nội dung:

- Kiểu dữ liệu mảng 2 chiều

Bài 4.1 Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm m hàng, n cột ($1 < m, n \leq 10$), mỗi phần tử là một số nguyên;
- In ma trận vừa nhập ra màn hình;
- Tính và in ra màn hình tổng tất cả các phần tử trong ma trận;
- Tìm và in ra phần tử có giá trị lớn nhất trong ma trận;
- Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 4.2 Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm mxn số thực ($1 < m, n \leq 10$);
- In ma trận vừa nhập ra màn hình;
- Đếm và in ra màn hình số lượng phần tử có giá trị x trong ma trận, với x được nhập vào từ bàn phím;
- Cho phép người dùng nhập hai chỉ số hàng (i1, i2), sau đó hoán đổi các phần tử tương ứng trên hai hàng này với nhau.
- Tính và in ra màn hình tổng các phần tử trong từng hàng;
- Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 4.3 Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm nxn phần tử là số nguyên ($3 \leq n \leq 10$);
- In ma trận vừa nhập ra màn hình;
- In ra các phần tử là số chẵn trong ma trận vừa nhập, phần tử là số lẻ sẽ in *

Ví dụ: Cho ma trận 3x3 như dưới đây => kết quả in ra

4	6	5	4	6	*
2	8	4	2	8	4
7	1	2	*	*	2

- In ra các phần tử nằm bên trên đường chéo chính (bao gồm cả đường chéo chính)

Ví dụ: Cho ma trận 3x3 như dưới đây => kết quả in ra

1	2	3	1	2	3
4	5	6		5	6
7	8	9			9

- Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 4.4 Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm nxn phần tử là số thực ($2 \leq n \leq 10$);
- In ma trận vừa nhập ra màn hình;
- Tính và in ra màn hình trung bình cộng các phần tử nằm trên đường chéo chính trong ma trận;
- Kiểm tra ma trận có toàn số dương hay không;

- Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 4.5* Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm m hàng, n cột ($2 < m, n \leq 10$), mỗi phần tử là một số nguyên;
- In ma trận vừa nhập ra màn hình;
- Tính và in ra màn hình tổng tất cả các phần tử nằm trên đường biên ngoài của ma trận;

Ví dụ: Cho ma trận 3x4 như bên dưới => Tổng các phần tử trên biên: $1 + 2 + 3 + 4 + 5 + 8 + 9 + 10 + 11 + 12 = 65$

1 2 3 4

5 6 7 8

9 10 11 12

- Tìm và in ra phần tử có giá trị lớn nhất trong cột thứ p của ma trận, với p được nhập vào từ bàn phím;
- Cho phép người dùng nhập hai chỉ số cột (j1, j2), sau đó hoán đổi các phần tử tương ứng trên hai cột này với nhau.
- Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 4.6* Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm nxn phần tử là số nguyên ($2 \leq n \leq 10$);
- In ma trận vừa nhập ra màn hình;
- In ra các phần tử là số lẻ trong ma trận vừa nhập, phần tử là số chẵn sẽ in ‘?’

Ví dụ: Cho ma trận 3x3 như dưới đây => kết quả in ra

4 6 5

? ? 5

2 8 4

? ? ?

7 1 2

7 1 ?

- In ra các phần tử nằm bên dưới đường chéo phụ (bao gồm cả đường chéo phụ)

Ví dụ: Cho ma trận 3x3 như dưới đây => kết quả in ra

1 2 3

3

4 5 6

5 6

7 8 9

7 8 9

- Tính và in ra tổng của các phần tử có giá trị chẵn thuộc đường chéo chính;
- Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 4.7* Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm nxn phần tử là số nguyên ($3 \leq n \leq 10$);
- In ma trận vừa nhập ra màn hình;
- Kiểm tra ma trận có phải là ma trận đơn vị hay không ($A[i][j] == 1$ nếu $i == j$, còn lại $A[i][j] == 0$);
- Kiểm tra ma trận có đối xứng hay không;
- Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 4.8* Viết chương trình thực hiện các yêu cầu sau:

- Nhập vào ma trận gồm $m \times n$ số thực ($1 < m, n \leq 10$);
- In ma trận vừa nhập ra màn hình;
- In ra màn hình ma trận chuyển vị;
- Xóa hàng thứ i trong ma trận, với i được nhập vào từ bàn phím;
- Hàm `main()` gọi các hàm trên thực hiện để kiểm tra kết quả.