

CSC10001

## LAB04 – MẢNG

FIT-HCMUS

### Bài tập 1    Mảng 1 chiều

1. Hàm nhập mảng n phần tử  
`void setArray(int a[], int &n);`
2. Hàm xuất mảng n phần tử  
`void printArray(int a[], int n);`
3. Hàm tính tổng các phần tử của mảng  
`int sumArray(int a[], int n);`
4. Hàm tìm min của mảng  
`int findMin(int a[], int n);`
5. Hàm đếm mảng có bao nhiêu số nguyên tố  
`int checkPrime(int n);`  
`int countNumberOfPrime(int a[], int n);`
6. Hàm kiểm tra mảng có tăng dần không  
`int isAscendingOrder(int a[], int n);`
7. Hàm đếm số lần xuất hiện của giá trị x  
`int countOccurrences(int a[], int n, int x);`
8. Hàm xoá các phần tử có giá trị x  
`void deleteItems(int a[], int &n, int x);`
9. Hàm xoá no\_items phần tử liên tục bắt đầu từ vị trí index cho trước  
`void deleteItemsFromIndex(int a[], int &n, int index, int no_items);`
10. Hàm nối 2 mảng thành 1 mảng  
`void concatTwoArrays(int a[], int n, int b[], int m, int c[]);`
11. Hàm nối 2 mảng xen kẽ thành 1 mảng (c = a[0], b[0], a[1], b[1], ...)  
`void concatInterspersedlyTwoArrays(int a[], int n, int b[], int m, int c[]);`

## Bài tập 2 Mảng ký tự / Chuỗi

Dùng theo 2 cách: mảng char tự định nghĩa hoặc dùng thư viện `string`.

Chuỗi có chiều dài `n`

1. Hàm xuất chuỗi theo thứ tự ngược  
`void reverseCharArray(char a[MAX])`
2. Hàm kiểm tra chuỗi đối xứng  
`int checkSymmetricArray(char a[MAX]);`
3. Hàm đếm số lần xuất hiện của ký tự `c`  
`int countOccurrencesChar(char a[MAX], char c);`
4. Hàm tìm ký tự xuất hiện nhiều nhất chuỗi và số lần nó xuất hiện  
`int findMostFrequentChar(char a[MAX], char &chr);`
5. Hàm chèn ký tự vào chuỗi tại vị trí `position`  
`void insertCharAtPosition(char str[], char chr, int pos);`
6. Nhập chuỗi số nguyên, xuất chuỗi có dấu , ngăn cách giữa triệu, ngàn  
`void formatNumberWithCommas(char str[]);`

## Bài tập 3 Mảng 2 chiều

1. Hàm nhập mảng `m` dòng `n` cột  
`void set2dArray(int a[][MAX], int &m, int &n);`
2. Hàm xuất mảng `m` dòng `n` cột  
`void print2dArray(int a[][MAX], int m, int n);`
3. Hàm tính tổng các phần tử của mảng  
`int sumArray(int a[][MAX], int m, int n);`
4. Hàm tính tổng đường chéo chính  
`int sumDiagonal(int a[][MAX], int m, int n);`
5. Hàm tìm min của mảng  
`int findMin(int a[][MAX], int m, int n);`
6. Hàm đếm mảng có bao nhiêu phần tử là số nguyên tố  
`int countNumberOfPrime(int a[][MAX], int m, int n);`
7. (\*) Hàm đếm mảng có bao nhiêu số nguyên tố khác nhau  
`int countNumberOfUniquePrime(int a[][MAX], int m, int n);`
8. Hàm đếm số lần xuất hiện của giá trị `x`  
`int countOccurrences(int a[][MAX], int m, int n, int x);`
9. Hàm xóa các dòng và cột có chứa giá trị `x`  
`void deleteItems(int a[][MAX], int &m, int &n, int x);`

## Bài tập 4 Bài tập tổng hợp

1. (\*) Viết chương trình cho phép người dùng nhập vào tên của  $n$  ứng cử viên trong một cuộc bầu cử địa phương và số phiếu mà mỗi ứng cử viên nhận được. Sau đó, chương trình sẽ xuất ra từng tên ứng cử viên, số phiếu nhận được và tỷ lệ phần trăm của tổng số phiếu mà ứng cử viên nhận được. Chương trình của bạn cũng nên xuất ra người chiến thắng trong cuộc bầu cử.

Một ví dụ đầu ra:

Candidate	Votes	Received % of Total Votes
Hoa	5000	25.91
Thang	4000	20.73
Thanh	6000	31.09
Hang	2500	12.95
Tuan	1800	9.33
Total	19300	
The Winner of the Election is Thanh.		

2. (\*) Viết chương trình sử dụng mảng hai chiều để lưu trữ cao nhất và nhiệt độ thấp nhất cho mỗi tháng trong năm. Đầu ra của chương trình là: trung bình nhiệt độ cao, trung bình nhiệt độ thấp, và nhiệt độ cao nhất và thấp nhất trong năm. Chương trình của bạn phải bao gồm các chức năng sau:

- Hàm `getData`: giúp lấy và lưu trữ dữ liệu trong mảng hai chiều.
- Hàm `averageHigh`: tính toán và trả về trung bình nhiệt độ cao của 12 tháng trong năm.
- Hàm `averageLow`: tính toán và trả về trung bình nhiệt độ thấp của 12 tháng trong năm.
- Hàm `indexHighTemp`: trả về vị trí của nhiệt độ cao nhất trong mảng.
- Hàm `indexLowTemp`: trả về vị trí của nhiệt độ thấp nhất trong mảng.

3. (\*) Viết chương trình có thể được sử dụng để chỉ định chỗ ngồi cho máy bay thương mại. Máy bay có 13 hàng, với sáu hàng ghế mỗi hàng. Hàng 1 và 2 là hạng nhất, hàng 3 đến 7 là hạng thương gia và hàng 8 đến 13 là hạng phổ thông. Chương trình của bạn phải cho phép người dùng nhập thông tin sau:

- a. Loại vé (hạng nhất, hạng thương gia hoặc hạng phổ thông)
- b. Ghế mong muốn

Hiển thị sơ đồ chỗ ngồi theo mẫu sau:

	A	B	C	D	E	F
Row 1	*	*	X	*	X	X
Row 2	*	X	*	X	*	X
Row 3	*	*	X	X	*	X
Row 4	X	*	X	*	X	X
Row 5	*	X	*	X	*	*
Row 6	*	X	*	*	*	X
Row 7	X	*	*	*	X	X
Row 8	*	X	*	X	X	*
Row 9	X	*	X	X	*	X
Row 10	*	X	*	X	X	X
Row 11	*	*	X	*	X	*
Row 12	*	*	X	X	*	X
Row 13	*	*	*	*	X	*

Ở đây, \* cho biết chỗ ngồi còn trống; X cho biết chỗ ngồi đã có người đặt. Làm một chương trình menu điều khiển cho phép hiển thị sơ đồ chỗ ngồi của máy bay như trên và cho phép người dùng đưa ra các lựa chọn phù hợp.