

LAB 3 - TÌM KIẾM

Tất cả các thao tác của các bài sau đây đều phải có menu để cho phép người dùng chọn

Bài 1:

Nhập mảng 1 chiều có N phần tử nguyên và thực hiện các thao tác sau đây:

- Xuất mảng 1 chiều nói trên ra màn hình.
- Viết hàm tìm kiếm giá trị x được nhập vào từ bàn phím trong mảng và trả về kết quả tìm thấy hay không (yêu cầu sử dụng thuật toán tìm kiếm tuyến tính và tìm kiếm tuyến tính cải tiến).
- Viết hàm tìm vị trí chứa giá trị nhỏ nhất trong mảng.
- Viết hàm tìm tất cả các vị trí có giá trị lớn nhất trong mảng.

Bài 2: Phát sinh ngẫu nhiên một mảng N phần tử theo thứ tự tăng dần sau đó thực hiện các thao tác sau:

- Xuất mảng 1 chiều nói trên ra màn hình.
- Viết hàm tìm vị trí giá trị x được nhập vào từ bàn phím trong mảng bằng thuật toán tìm kiếm tuyến tính.
- Viết thêm hàm tìm kiếm bằng nội suy theo yêu cầu của câu B sau đó so sánh (in ra) thời gian thực hiện của phương pháp tìm kiếm bằng nội suy và tìm kiếm tuyến tính.

Gợi ý: Tạo mảng tăng dần và tính thời gian thực thi chương trình

```
#include <iostream>
#include <cstdlib> // for srand() and rand()
#include <ctime> // for time()
using namespace std;

// định nghĩa số phần tử mảng
#define MAX 1000

// khai báo prototype
void nhapMang(int arr[], int &n);
void xuatMang(int arr[], int n);

int main()
{
    clock_t tStart = clock(); // Bắt đầu tính thời gian chạy chương trình
    int myArray[MAX]; // mảng myArray có MAX phần tử
    int nSize; // nSize là số phần tử được sử dụng, do user nhập
```

```

        // nhập xuất mảng tự động
nhapMang(myArray, nSize);
xuatMang(myArray, nSize);
// Thời gian chạy chương trình bằng thời gian kết thúc trừ thời gian đầu
cout<<"Time taken: "<< (double)(clock() - tStart) / CLOCKS_PER_SEC;

system("pause");

return 0;
}

// hàm nhập mảng
void nhapMang(int arr[], int &n)
{
    // khởi tạo số ngẫu nhiên
    srand(time(NULL));

    cout << "Nhap so luong phan tu n : ";
    cin >> n;

    if (n > 0)
    {
        arr[0] = rand() % 100;
        // khởi tạo ngẫu nhiên từng phần tử từ chỉ số 1 đến n - 1
        for (int i = 1; i < n; i++)
        {
            arr[i] = arr[i - 1] + rand() % 100;
        }
    }
}

// hàm xuất mảng
void xuatMang(int arr[], int n)
{
    // xuất từng phần tử cho mảng từ chỉ số 0 đến n - 1
    for (int i = 0; i < n; i++)
    {
        cout << "arr[" << i << "] = " << arr[i] << endl;
    }
}

```