

# Hướng dẫn thực hành 5

## Mục tiêu

---

- Khai báo và khởi tạo các phần tử trong array.
- Nhập dữ liệu vào array và in dữ liệu lưu trong array.
- Tìm kiếm, thêm, xóa, và thực hiện một số tính toán với dữ liệu được lưu trong array.
- Truyền dữ liệu được lưu trữ trong array cho hàm (bằng tham chiếu và hằng tham chiếu).

## Hướng dẫn

---

### Bài 1.

Chương trình C++

```
//Bài thực hành 5 - câu 1
#include <iostream>
#include <iomanip>
using namespace std;
```

```
int main()
```

```
{
```

```
    double alpha[50];
```

khai báo mảng

```
    for (int i = 0; i <= 24; i++)
```

```
        alpha[i] = i*i;
```

```
    for (int i = 25; i <= 49; i++)
```

```
        alpha[i] = i*i*i;
```

lưu dữ liệu vào mảng

```
    for (int i = 0; i <= 49; i++)
```

```
    {
```

```
        cout << setw(8) << alpha[i];
```

```
        if ((i + 1) % 10 == 0) cout << endl;
```

```
    }
```

in dữ liệu trong mảng

```
    return 0;
```

```
}
```

### Bài 2.

Chương trình C++

```
//Bài thực hành 5 - câu 2
#include <iostream>
using namespace std;
```

```
int main()
```

```
{
```

```
    const int MAXSIZE = 100;    //tôi đã 100 phần tử
```

```
    double values[MAXSIZE];
```

```
    int currentSize = 0;
```

```
    //số phần tử hiện tại
```

khai báo mảng

```

cout << "Nhap so phan tu cua mang: ";
cin >> currentSize;

cout << "Nhap vao " << currentSize << " phan tu." << endl;
for (int i = 0; i <= currentSize - 1; i++)
{
    cout << "Phan tu thu " << i + 1 << ": ";
    cin >> values[i];
}
cout << endl;

cout << "Cac gia tri da nhap vao mang la: ";
for (int i = 0; i <= currentSize - 1; i++)
{
    cout << values[i] << " ";
}
cout << endl;
return 0;
}

```

lưu dữ liệu vào mảng

in dữ liệu trong mảng

### Bài 3. Tính tổng và giá trị trung bình của các phần tử trong mảng.

Tham khảo chương 5, slide 17.

### Bài 4. Tìm phần tử lớn nhất trong mảng.

Tham khảo chương 5, slide 22.

### Bài 5. Xóa phần tử tại vị trí p trong mảng a có n phần tử.

Trường hợp 1: Không cần đảm bảo trật tự của các phần tử sau khi xóa.

1. Gán phần tử tại vị trí  $n - 1$  vào vị trí p
2. Giảm số phần tử đi 1.

Trường hợp 2: Đảm bảo trật tự của các phần tử sau khi xóa.

1. Dời các phần tử từ vị trí p + 1 sang trái (hoặc lên trên) một vị trí.
2. Giảm số phần tử đi 1.

### Chương trình C++

```

//Bài thực hành 5 - câu 5 - xóa phần tử tại vị trí trong mảng
#include <iostream>
using namespace std;

int main()
{
    const int MAXSIZE = 100;
    int values[MAXSIZE]; //mang luu toi da 100 so nguyen
    int currentSize = 0; //so phan tu hien dang luu trong mang
    int pos = 0; //vi tri xoa

    cout << "Nhap so phan tu cua mang: ";
    cin >> currentSize;
}

```

```

    cout << "Nhap vao " << currentSize << " phan tu." << endl;
    for (int i = 0; i <= currentSize - 1; i++)
    {
        cout << "Phan tu thu " << i + 1 << ": ";
        cin >> values[i];
    }
    cout << endl;

    cout << "Nhap vi tri can xoa (vi tri thu nhat la 1): ";
    cin >> pos;
    if (pos < 1 || pos > currentSize)
    {
        cout << "Vi tri xoa khong hop le!" << endl;
        return 1;
    }

    for (int i = pos - 1; i <= currentSize - 2; i++)
        values[i] = values[i + 1]; //doi phan tu sang trai

    currentSize = currentSize - 1; //giam so phan tu cua mang di 1

    cout << "Mang co " << currentSize << " phan tu: ";
    for (int i = 0; i <= currentSize - 1; i++)
    {
        cout << values[i] << " ";
    }
    cout << endl;

    return 0;
}

```

## Bài 6. Viết hàm nhập, xuất mảng

### Chương trình C++

```

//Bài thực hành 5 - câu 6
#include <iostream>
using namespace std;

void nhap(int arr[], int arrSize);
void xuat(const int arr[], int arrSize);

int main()
{
    const int MAXSIZE = 100; //toi da 100 phan tu
    int a[MAXSIZE];
    int n = 0; //so phan tu hien thoi
    int p = 0; //vi tri xoa
    int x = 0; //phan tu can tim

    cout << "Nhap so phan tu cua mang: ";
    cin >> n;
    nhap(a, n);

    cout << "Cac phan tu trong mang: ";
    xuat(a, n);

    return 0;
}

```

```

void nhap(int arr[], int arrSize)
{
    cout << "Nhập vào " << arrSize << " phần tử." << endl;
    for (int i = 0; i <= arrSize - 1; i++)
    {
        cout << "Phần tử thứ " << i + 1 << ": ";
        cin >> arr[i];
    }
    cout << endl;
}

void xuat(const int arr[], int arrSize)
{
    for (int i = 0; i <= arrSize - 1; i++)
    {
        cout << arr[i] << " ";
    }
    cout << endl;
}

```

Sinh viên tự định nghĩa các hàm tìm kiếm và xóa.