**Bài tập**

Cho mảng số nguyên n phần tử  arr được nhập từ bàn phím, bạn hãy viết hàm trả về phần tử lớn nhất trong mảng arr sử dụng con trỏ và hiển thị ra màn hình:

Max value = {P}

Với {P} là giá trị của phần tử lớn nhất trong mảng.

**Lý thuyết**

Bài này sẽ giúp bạn hiểu hơn về mảng.

Tên mảng (chính là biến arr trong bài này) mà từ trước tới nay bạn sử dụng có bản chất giống như một con trỏ. Để hiểu rõ hơn bạn có thể xem ví dụ sau:

#include <iostream>

using namespace std;

int main() {

int arr[] = { 1, 2, 3, 4, 5 };

// Hiển thị giá trị của arr

cout << arr << endl;

// Hiển thị địa chỉ của phần tử đầu tiên trong mảng

cout << &arr[0];

return 0;

}

Kết quả khi chạy chương trình

0039FA9C

0039FA9C

Có thể thấy giá trị của arr chính là địa chỉ của phần tử đầu tiên trong mảng (arr trỏ tới địa chỉ của arr[0]), nên có thể coi arr giống như 1 con trỏ.

Lưu ý: arr không phải là một con trỏ mà arr được dùng giống như một con trỏ.

Bạn có thể duyệt mảng bằng biến arr mà biến arr giống như một con trỏ nên bạn cũng có thể duyệt mảng bằng 1 con trỏ. Xem ví dụ sau:

#include <iostream>

using namespace std;

int main() {

int arr[] = { 1, 2, 3, 4, 5 };

int\* p = &arr[0];

for (int i = 0; i < 5; i++) {

cout << p[i] << " ";

}

return 0;

}

Kết quả khi chạy chương trình:

1 2 3 4 5

Có thể thấy bạn có thể sử dụng con trỏ p giống như sử dụng biến arr. Đọc tới đây bạn đã biết cách sử dụng con trỏ để duyệt mảng, hãy quay lại phần bài tập và làm thử.

**Hướng dẫn**

Code mẫu:

#include <iostream>

using namespace std;

int getMaxValue(int\* p, int n) {

int maxValue = p[0];

for (int i = 0; i < n; i++) {

if (p[i] > maxValue) {

maxValue = p[i];

}

}

return maxValue;

}

int main() {

int n;

int arr[100];

cin >> n;

for (int i = 0; i < n; i++) {

cin >> arr[i];

}

cout << "Max value = " << getMaxValue(&arr[0], n);

return 0;

}

Bạn có thể truyền thẳng biến arr vào hàm getMaxValue() thay vì truyền &arr[0] vì arr chính là &arr[0]:

#include <iostream>

using namespace std;

int getMaxValue(int\* p, int n) {

int maxValue = p[0];

for (int i = 0; i < n; i++) {

if (p[i] > maxValue) {

maxValue = p[i];

}

}

return maxValue;

}

int main() {

int n;

int arr[100];

cin >> n;

for (int i = 0; i < n; i++) {

cin >> arr[i];

}

cout << "Max value = " << getMaxValue(arr, n);

return 0;

}