

Bài tập Thực hành Tuần 2: Con trỏ và cấp phát bộ nhớ động

Bài 1: Cho chương trình sau

```
/* Con trỏ trong C/C++ */
#include <iostream>
using namespace std;

int main() {
    int var = 10;
    // Khai báo biến con trỏ ptr1 và gán bởi địa chỉ biến var.
    int *ptr1 = &var;
    *ptr1 = 20;

    int **ptr2 = &ptr1;
    **ptr2 = 30; // (1)

    int ***ptr3 = &ptr2;
    ***ptr3 = 50; // (2)
    cout << "\nvar = " << var << endl;

    return 0; }
```

Cho biết kết quả chạy của chương trình? Giải thích tại sao có kết quả đó?

Bài 2: Viết hàm tìm tất cả các vị trí của x xuất hiện trên mảng, sử dụng con trỏ.

Ex.

Input: A[5] = {5, 10, 15, 5, 5}, x = 5

Output: Các vị trí xuất hiện của x: {0, 3, 4}.

Bài 3: Viết hàm tìm tất cả các vị trí của các phần tử trên mảng gần x nhất.

Ex.

Input: A[5] = {5, 10, 15, 10, 5}, x=11

Output: Các vị trí của các phần tử trên mảng A gần x nhất là: {1, 3}.

Bài 4 Viết chương trình quản lý sinh viên, với thông tin sinh viên gồm: name (C-string),

Id (C-string), Marks là mảng gồm 3 cột điểm: bài tập, thi giữa kỳ, và thi cuối kỳ.

Biết rằng công thức tính điểm trung bình như sau:

$$DTB = baitap*0.25 + giuaky*0.25 + cuoiky*0.5$$

Cấu trúc sinh viên được định nghĩa như sau:

```
typedef struct {
    char id[9]; // mã số
    char name[20];
    float Marks[3];
    float DTB; // Điểm trung bình
}Student;
```

a) Viết hàm xếp loại của sinh viên.

```
void xeploai(Student S);
```

Biết rằng:

- Nếu DTB < 5 : "Yeu";
- Nếu 5 <= DTB < 7: "Trung binh";
- Nếu 7 <= DTB < 8: "Kha";
- Nếu DTB >= 8: "Gioi".

b) Sử dụng mảng động để cài đặt danh sách sinh viên, Viết hàm nhập danh sách sinh viên.

c) Viết hàm xuất danh sách sinh viên có thông tin xếp loại.

d) Viết hàm thêm một sinh viên vào danh sách.

e) Viết hàm xóa các sinh viên có điểm trung bình nhỏ hơn 5, và xuất danh sách sau khi xóa.

Sinh viên có thể tham khảo hàm nhập sinh viên sau:

```
void nhap_sv(Student &sv) {  
    cout << "Nhap id: ";  
    cin.get(sv.id, 9);  
    /*Clearing the input buffer*/  
    cin.ignore();  
    cout << "Nhap name: ";  
    cin.get(sv.name, 20);  
    cout << "Marks: baitap, giuaky, cuoiky: ";  
    for (int i = 0; i < 3; i++){  
        cin >> sv.Marks[i];  
    }  
    sv.DTB = sv.Marks[0] * 0.25 + sv.Marks[1] * 0.25 + sv.Marks[2]*0.5;
```

```
};
```
