1. Giải thích kết quả

#include <iostream>

using namespace std;

typedef int \*IntPtrType; Định danh kiểu con trỏ IntPtrType

int main()

{

IntPtrType ptr\_a, ptr\_b, \*ptr\_c; // khai báo con trỏ ptr\_a, ptr\_b

// ptr\_c: khai báo con trỏ trỏ tới vùng nhớ của con trỏ

ptr\_a = new int; // cấp phát vùng nhớ 4 bytes và cho con trỏ ptr\_a trỏ đến

\*ptr\_a = 3; // gán giá trị vùng nhớ nơi con trỏ ptr\_a trỏ tới = 3

ptr\_b = ptr\_a; // cho con trỏ b trỏ tới cùng vùng nhớ con trỏ a đang trỏ tới

cout << \*ptr\_a << " " << \*ptr\_b << "\n";

// xuất giá trị vùng nhớ con trỏ a, b = 3

ptr\_b = new int; // cấp phát vùng nhớ mới 4 bytes và cho con trỏ ptr\_b trỏ tới

\*ptr\_b = 9; // gán giá trị vùng nhớ nơi con trỏ ptr\_b trỏ tới = 9

cout << \*ptr\_a << " " << \*ptr\_b << "\n"; // xuất giá trị vùng nhớ con trỏ ptr\_a và ptr\_b đang trỏ tới là 3 và 9

\*ptr\_b = \*ptr\_a; // =3

// gán giá trị vùng nhớ ptr\_b trỏ tới = giá trị vùng nhớ ptr\_a trỏ tới

cout << \*ptr\_a << " " << \*ptr\_b << "\n";

delete ptr\_a; // giải phóng liên kết con trỏ a

ptr\_a = ptr\_b; // cho con trỏ a trỏ tới vùng nhớ con trỏ b đang trỏ tới

cout << \*ptr\_a << " " << \*ptr\_b << "\n";

// xuất giá trị vùng nhớ con trỏ a, b đang trỏ tới = 3

ptr\_c = &ptr\_a; // con trỏ c được gán = địa chỉ của con trỏ a

cout << \*ptr\_c << " " << \*\*ptr\_c << "\n";

// \*ptr\_c: địa chỉ con trỏ a, \*\*ptr\_c giá trị vùng nhớ con trỏ a đang trỏ tới

delete ptr\_a; // giải phóng liên kết của con trỏ a

ptr\_a = NULL; // cho con trỏ a trỏ tới NULL

system("pause");

return 0;

}

2.

#include​ ​<iostream>  
using​ ​namespace​ std;  
void​ main()  
{  
 int​ a[4] = {1, 2, 3, 4};  
 int​ \*p = a;  
 int​ \*p2 = ​new​ ​int​;  
 // delete​ p; không thể giải phóng vùng nhớ mảng tĩnh   
 // delete​ a;  
 delete​ p2;  
}

3.1 Vùng nhớ được cấp phát trên Heap sẽ không tự động hủy bởi chương trình khi kết thúc khối lệnh, việc thu hồi vùng nhớ đã cấp phát trên Heap được giao cho lập trình viên tự quản lý. Nếu trong chương trình có yêu cầu cấp phát bộ nhớ trên Heap mà không được thu hồi hợp lý sẽ gây lãng phí tài nguyên hệ thống

3.2 Khi không muốn sử dụng tiếp vùng nhớ đã được cấp phát cho chương trình trên Heap, chúng ta nên trả lại vùng nhớ đó cho hệ điều hành.

3.3 Delete dùng để giải phóng 1 con trỏ đã được cấp phát trước đó, còn nếu đó là 1 mảng thì dùng delete[]

Demo: **int** \*a = **new** **int**;

**delete** a;

**char** \*arr = **new** **char**[5];

**delete**[] arr;

4.

1. \*p1 == 10  
2. \*p2 == 0x6ffe14 // địa chỉ con trỏ p1  
3. \*(\*p2) == 10