Ví dụ 1: Con trỏ và Đối tượng (cấp phát tĩnh, động)



#include <stdio.h>

#include <stdlib.h>

int main()

{

    /\* Con trỏ tham chiếu tới đối tượng tĩnh \*/

    int n; // biến số nguyên n

    // biến con trỏ chứa địa chỉ tham chiếu tới vùng nhớ chứa số nguyên

    int \*p = NULL; // con trỏ không sử dụng phải gán NULL

    n = 20;

    p = &n;                                // gán địa chỉ của biến n cho

    printf("n: Addr=%X, Val=%d\n", &n, n); // in thông tin biến n: địa chỉ, giá trị

    // địa chỉ, giá trị, giá trị tham chiếu

    printf("p: Addr=%X, Val=%X, ValRef=%d\n", &p, p, \*p);

    /\* Con trỏ tham chiếu tới đối tượng động \*/

    int \*pn = NULL;

    // cấp phát vùng có kích thước kiểu int

    // và pn chứa địa chỉ vừa cấp phát

    pn = (int \*)malloc(sizeof(int));

    (\*pn) = 10;                                       // gán 10 vào đối tượng pn tham chiếu

    printf("sizeof(int): %d byte(s)\n", sizeof(int)); // in kích thước kiểu int

    // địa chỉ, giá trị, giá trị tham chiếu

    printf("pn: Addr=%X, Val=%X, ValRef=%d\n", &pn, pn, \*pn);

    // ! hủy vùng nhớ sau khi sử dụng xong

    free(pn);

    return 0;

}

Ví dụ 2: Truyền tham số cho Con trỏ và Đối tượng



#include <stdio.h>

#include <stdlib.h>

// [1] Truyền tham trị (Pass by value)

void HoanVi\_ThamTri(int a, int b)

{

    int tmp = a;

    a = b;

    b = tmp;

}

// [2] Truyền tham chiếu bằng con trỏ (Pass by pointer)

void HoanVi\_ThamChieu(int \*a, int \*b)

{

    int tmp = (\*a);

    (\*a) = (\*b);

    (\*b) = tmp;

}

int main()

{

    int x = 5, y = 10;

    // Trước khi hoán đổi

    printf("Trước khi hoán đổi: x=%d, y=%d\n", x, y);

    // Truyền tham trị (không thay đổi giá trị của x và y trong main)

    HoanVi\_ThamTri(x, y);

    printf("Sau khi hoán đổi (tham trị): x=%d, y=%d\n", x, y);

    // Truyền tham chiếu bằng con trỏ (thay đổi giá trị của x và y trong main)

    HoanVi\_ThamChieu(&x, &y);

    printf("Sau khi hoán đổi (tham chiếu): x=%d, y=%d\n", x, y);

    return 0;

}

Ví dụ 3: Vẽ mô hình bộ nhớ và giải thích cho tình huống: con trỏ tham chiếu tới mảng

tĩnh, con trỏ quản lý mảng động



#include <stdio.h>

#include <stdlib.h>

int main()

{

    /\* Con trỏ với mảng tĩnh \*/

    int a[5] = {2, 4, 5, 8}; // Mảng tĩnh

    int na = 3;              // Số phần tử của mảng

    int \*p = NULL;           // con trỏ rỗng

    p = a + 1;               // hay &a[1] - trỏ tới phần tử thứ 1

    // địa chỉ, giá trị, giá trị tham chiếu

    printf("p: Addr=%X, Val=%X, ValRef=%d\n", &p, p, \*p);

    p = p + 1;     // p = p + 1\*sizeof(int) --> di chuyen qua ben phai 1 don vi

    int x = p[-2]; // \*(p - 2) - di chuyen qua trai 2 don vi

    /\* Con trỏ với mảng động \*/

    int \*b = NULL; // Khai báo mảng rỗng

    int nb = 0;

    nb = 3;

    b = (int \*)malloc(sizeof(int) \* nb); // stdlib.h, Cấp phát nb phần tử (C)

    // b = new int[nb]; // Cấp phát nb phần tử (C++)

    \*(b + 0) = 10;

    \*(b + 1) = 20;

    b[2] = 30;

    if (b != NULL)

        free(b); // Hủy vùng nhớ cấp phát

    return 0;

}

Ví dụ 4: Vẽ mô hình bộ nhớ và giải thích cho tình huống: truyền mảng như tham trị,

truyền mảng như tham chiếu, truyền mảng như tham biến



#include <stdio.h>

#include <stdlib.h>

#define MAX 100 // so luong phan tu toi da cua day

/\*

Nhap day so nguyen tu ban phim

+ Vao: ban phim

+ Ra : (\*a), n

+ Vi du: (\*a) = {1,2,3}, n = 3

\*/

void NhapMang(int \*&a, int &n) // x <-> a, nx <-> n // truyen tham bien

{

    printf("Moi ban nhap so luong phan tu: "); // nhap n

    scanf("%d", n);

    a = new int[n];                  // cap phat mang a co n phan tu

    for (int i = 0; i <= n - 1; i++) // duyet tu vi tri 0 den n-1

    {

        // nhap phan tu thu a[i]: (a+i) <-> &a[i]

        printf("Phan tu %d: ", i);

        scanf("%d", a + i); // scanf("%d",&a[i]);

    } // for i

} // NhapMang

void NhapMang(int \*\*a, int \*n) // truyen tham chieu

{

    printf("Moi ban nhap so luong phan tu: "); // nhap (\*n)

    scanf("%d", n);

    (\*a) = new int[(\*n)];               // cap phat mang (\*a) co (\*n) phan tu

    for (int i = 0; i <= (\*n) - 1; i++) // duyet tu vi tri 0 den (\*n)-1

    {

        // nhap \*((\*a)+i) <-> (\*a)[i]

        printf("Phan tu %d: ", i);

        scanf("%d", (\*a) + i); // scanf("%d",&(\*a)[i]);

    } // for i

} // NhapMang

/\*

Xuat day so nguyen ra man hinh

+ Vao: (\*a), n

+ Ra : man hinh + Vi du:

\*a = {2,5,3}, n = 3

=> man hinh = 2 5 3

\*/

void XuatMang(int \*a, int n) // truyen tham tri

{

    printf("Day so co %d phan tu: ", n); // xuat so luong phan tu

    for (int i = 0; i <= n - 1; i++)     // duyet tu vi tri 0 den n-1 lam

    {

        printf("%d ", \*(a + i)); // xuat a[i] hay printf("%d",a[i]);

    } // for i

    printf("\n"); // xuong dong

} // XuatMang

void DoiChan(int \*a, int n) // truyen tham tri, đổi các giá trị chẵn thành số 0

{

    for (int i = 0; i <= n - 1; i++)

    {

        if (\*(a + i) % 2 == 0)

        {

            \*(a + i) = 0;

        }

    }

}

int main()

{

    /\* Khai bao bien \*/

    int \*b = NULL, k = 0;

    /\* Nhap du lieu \*/

    NhapMang(&b, &k);

    // NhapMang(b,k);

    /\* Xuat du lieu \*/

    XuatMang(b, k);

    DoiChan(b, k);

    printf("\* Doi gia tri chan thanh cac so 0: \n:");

    XuatMang(b, k);

    /\* QUAN TRONG - Huy vung nho\*/

    if (b != NULL)

        delete[] b;

    /\* Dung man hinh xem ket qua \*/

    system("PAUSE");

    return 0;

} // main

Ví dụ 5: Cho dãy số nguyên a có n phần tử (n ≤ 100). Hãy tạo dãy b chứa các số chẵn của a và xuất b ra màn hình.



#include <stdio.h>

#include <stdlib.h>

#define MAX 100 // so luong phan tu toi da cua day

/\*

Nhap day so nguyen tu ban phim

+ Vao: ban phim

+ Ra : (\*a), n

+ Vi du: (\*a) = {1,2,3}, n = 3

\*/

void NhapMang(int \*\*a, int \*n)

{

    printf("Moi ban nhap so luong phan tu: "); // nhap (\*n)

    scanf("%d", n);

    while ((\*n) <= 0 || (\*n) > MAX) // kiem tra nhap (\*n)

    {                               // nhap lai (\*n)

        printf("Nhap sai! Nhap lai: ");

        scanf("%d", n);

    } // while n

    (\*a) = new int[(\*n)];               // cap phat mang (\*a) co (\*n) phan tu

    for (int i = 0; i <= (\*n) - 1; i++) // duyet tu vi tri 0 den (\*n)-1

    {

        // nhap \*((\*a)+i) <-> (\*a)[i]

        printf("Phan tu %d: ", i);

        scanf("%d", (\*a) + i); // scanf("%d",&(\*a)[i]);

    } // for i

} // NhapMang

/\*

Xuat day so nguyen ra man hinh

+ Vao: (\*a), n

+ Ra : man hinh + Vi du:

\*a = {2,5,3}, n = 3

=> man hinh = 2 5 3

\*/

void XuatMang(int \*a, int n)

{

    printf("Day so co %d phan tu: ", n); // xuat so luong phan tu

    for (int i = 0; i <= n - 1; i++)     // duyet tu vi tri 0 den n-1 lam

    {

        printf("%d ", \*(a + i)); // xuat a[i] hay printf("%d",a[i]);

    } // for i

    printf("\n"); // xuong dong

} // XuatMang

/\*

Dem so chan co tron day

+ Vao: \*a, n

+ Ra : dem

\*/

int DemChan(int \*a, int n)

{

    int dem = 0;

    for (int i = 0; i <= n - 1; i++)

    {

        if (\*(a + i) % 2 == 0)

            dem++;

    }

    return dem;

} // DemChan

/\*

Tach cac so chan

+ Vao: \*a, n

+ Ra : \*b, m

\*/

void TachChan(int \*a, int n, int \*\*b, int \*m)

{

    (\*m) = DemChan(a, n); // dem so luong phan tu day (\*b)

    (\*b) = new int[(\*m)]; // cap phat day so (\*b) co (\*m) phan tu

    int cs = 0;

    for (int i = 0; i <= n - 1; i++)

    {

        if (\*(a + i) % 2 == 0) // a[i]%2

        {

            \*((\*b) + cs) = \*(a + i); // (\*b)[cs] = a[i]

            cs++;

        }

    } // for

} // TachChan

int main()

{

    /\* Khai bao bien \*/

    int \*b = NULL, k = 0;

    int \*aChan = NULL, nChan;

    /\* Nhap du lieu \*/

    NhapMang(&b, &k);

    /\* Xu ly\*/

    TachChan(b, k, &aChan, &nChan);

    /\* Xuat du lieu \*/

    printf("+ Day chua cac so chan\n");

    XuatMang(aChan, nChan);

    /\* QUAN TRONG - Huy vung nho\*/

    delete[] b;

    delete[] aChan;

    /\* Dung man hinh xem ket qua \*/

    system("PAUSE");

    return 0;

} // main