



**ĐẠI HỌC KINH TẾ KỸ THUẬT CÔNG NGHIỆP**  
**KHOA CÔNG NGHỆ THÔNG TIN**

# **Bài Thực Tập 10**

## Lập trình với con trỏ

**Hà Nội – 2022**

# NỘI DUNG

1

**Mục tiêu bài học**

2

**Hướng dẫn học tập**

3

**Nội dung bài học**

4

**Giao nhiệm vụ tuần tiếp theo**

# MỤC TIÊU BÀI HỌC

- **Mục tiêu:**

Trang bị cho sinh viên kỹ năng lập trình cơ bản trong C++:

- + Hiểu về toán tử &, toán tử \*
- + Các phép toán với con trỏ
- + Cấp phát động
- + Viết chương trình sử dụng con trỏ
- + Con trỏ mảng, con trỏ chuỗi
- + Phát hiện và sửa lỗi

- **Kết quả đạt được:**

Sinh viên sử dụng thành thạo biến con trỏ, con trỏ mảng và con trỏ chuỗi để áp dụng giải các bài tập từ đơn giản đến phức tạp.

# HƯỚNG DẪN HỌC TẬP

**Để hoàn thành tốt bài học này sinh viên cần thực hiện những nhiệm vụ sau:**

- Đọc trước tài liệu: “B10\_Tailieu\_TTLTCB” phần nhắc lại kiến thức lý thuyết mục “E. TÓM TẮT LÝ THUYẾT”
- Cài đặt, sử dụng được công cụ thực hành Cfree 5.0.
- Thực hành trên máy tính các bài thực hành mẫu.
- Hoàn thành các bài thực hành tự làm cuối bài học.
- Hoàn thành các bài tập giao về nhà.
- Trao đổi, thảo luận với giảng viên qua các phương thức:
  - + Thảo luận đặt câu hỏi trên diễn đàn.

# NỘI DUNG BÀI HỌC

## I. Hướng dẫn ban đầu (90 phút)

- 1.1 Tổng hợp lý thuyết
- 1.2 Hướng dẫn thực hành bài số 01
- 1.3 Hướng dẫn thực hành bài số 02

## II. Hướng dẫn thường xuyên (90 phút)

- 2.1 Hướng dẫn thực hành bài số 03
- 2.2 Hướng dẫn thực hành bài số 04

## III. Bài tập tự giải (90 phút)

# I. HƯỚNG DẪN BAN ĐẦU

- Sinh viên cần cài đặt được công cụ thực hành.
- Yêu cầu sinh viên đọc lại và ghi nhớ các kiến thức lý thuyết trong mục E trong tài liệu “*B10\_Tailieu\_TTLTCB\_1*”
- Hướng dẫn chi tiết ví dụ mẫu để hiểu rõ về cách sử dụng con trỏ.

# 1.1 TỔNG HỢP LÝ THUYẾT

## Khai báo con trỏ

chỉ ra một kiểu cơ sở và một tên biến được đặt trước bởi dấu \*

Cú pháp:

```
type      *name ;
```

Ví dụ:

```
int      *a ;
```

# 1.1 TỔNG HỢP LÝ THUYẾT

## Khai báo con trỏ

Có thể khai báo một con trỏ trỏ đến bất kỳ kiểu dữ liệu nào.

```
int      *ptr_num;  
float    *ptr_x;
```

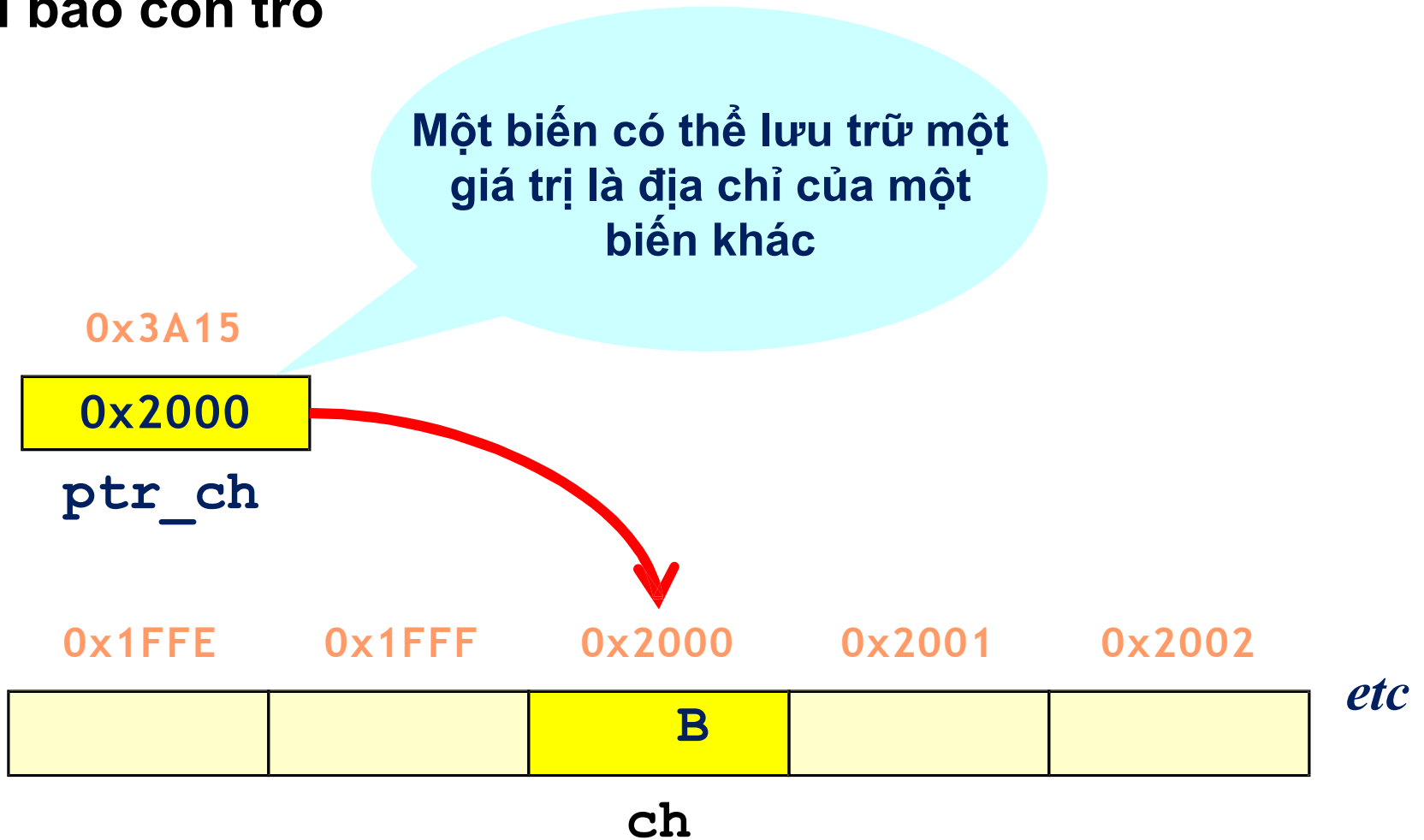
Một biến con trỏ luôn được khai báo với một toán tử \*

```
int *ptr_num1,  
    *ptr_num2; float  
    *ptr_x, *ptr_y;
```



# 1.1 TỔNG HỢP LÝ THUYẾT

## Khai báo con trỏ



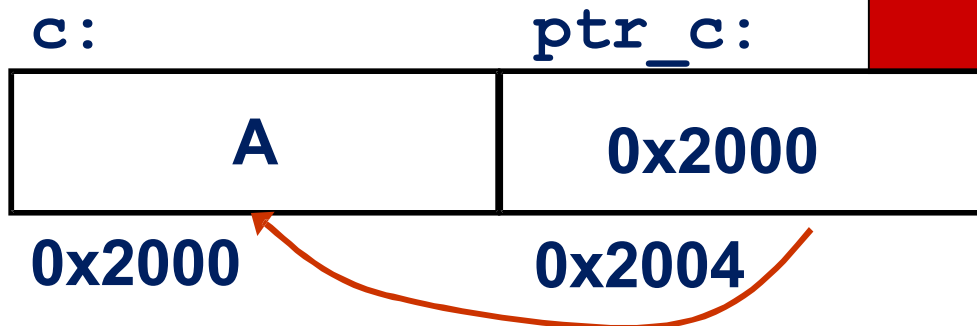
# 1.1 TỔNG HỢP LÝ THUYẾT

## Tham chiếu

Toán tử & được sử dụng để lấy địa chỉ tham chiếu của một con trỏ

```
char c = 'A';  
char *ptr_c;  
ptr_c = &c;
```

*Gán địa chỉ của **c**  
cho con trỏ **ptr\_c***



# 1.1 TỔNG HỢP LÝ THUYẾT

## Chú ý

Biến tham chiếu của một con trỏ phải có kiểu cùng kiểu với con trỏ.

```
int  aNum;  
  
char *ptr;  
  
ptr = &aNum;
```

Sai kiểu dữ  
liệu

# 1.1 TỔNG HỢP LÝ THUYẾT

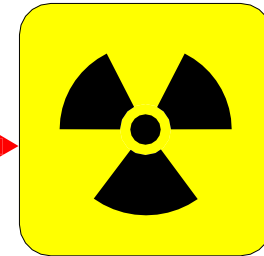
## Con trỏ **NULL**

```
int *ptr_num;
```

Hãy cẩn thận với con trỏ không được khởi tạo

???

ptr\_num



```
int *ptr_num = NULL;
```

NULL

ptr\_num

*Một con trỏ có tham chiếu NULL (không có địa chỉ)*

# 1.1 TỔNG HỢP LÝ THUYẾT

- Hai toán tử đặc biệt được sử dụng với con trỏ:

**&** và **\***

- **&** là toán tử một ngôi và nó trả về địa chỉ ô nhớ của toán hạng

```
ptr_x = &x;
```

- Toán tử **\*** là toán tử một ngôi và nó trả về giá trị chứa trong vùng nhớ được trỏ đến bởi biến con trỏ.

```
x = *ptr_x;
```

# 1.1 TỔNG HỢP LÝ THUYẾT

## Toán tử &

Cho biết địa chỉ bộ nhớ của một đối tượng

```
char ch = 'A';
```

0x2000

A

&ch

Trả về 0x2000

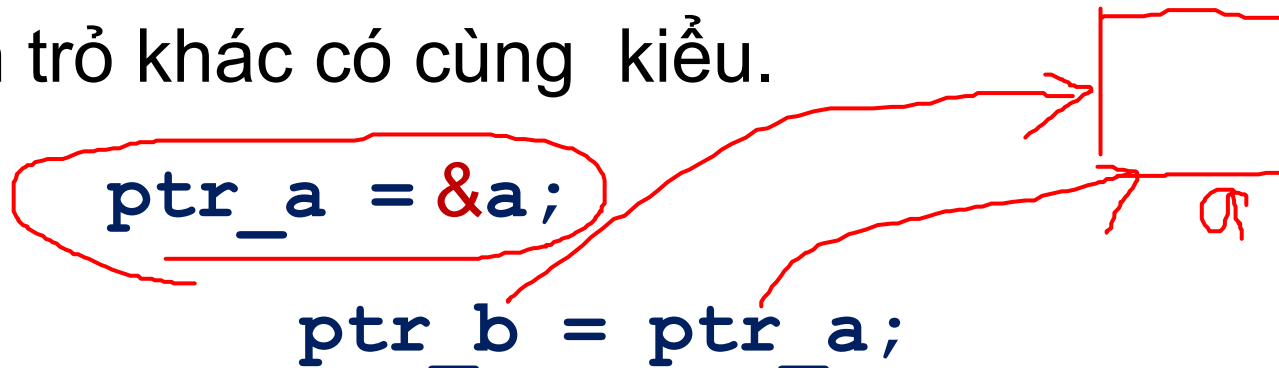
# 1.1 TỔNG HỢP LÝ THUYẾT

## Toán tử &

- Các giá trị có thể được gán cho con trỏ thông qua toán tử &.

```
ptr_a = &a;
```

- Ở đây địa chỉ của a được lưu vào biến ptr\_a.
- Cũng có thể gán giá trị cho con trỏ thông qua một biến con trỏ khác có cùng kiểu.



# 1.1 TỔNG HỢP LÝ THUYẾT

## Toán tử \*

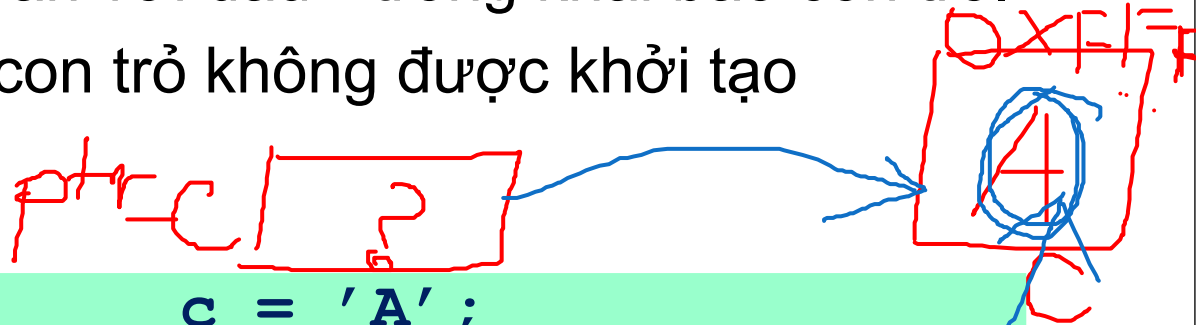
Chúng ta có thể sử dụng con trỏ để truy cập các biến mà chúng trỏ tới bởi toán tử \*.

- ✓ Không nên nhầm lẫn với dấu \* trong khai báo con trỏ.
- ✓ Hãy cẩn thận với con trỏ không được khởi tạo

Ví dụ:

```
char    c = 'A';  
char    *ptr_c = NULL;  
✓ ptr_c = &c;  
*ptr_c = 'B';
```

Thay đổi giá trị của **c**  
bởi con trỏ **ptr\_c**





# 1.1 TỔNG HỢP LÝ THUYẾT

## Toán tử \*

- Có thể gán giá trị cho các biến thông qua con trỏ

```
*ptr_a = 10;
```

- Câu lệnh trên gán giá trị 10 cho biến a nếu ptr\_a đang trỏ đến a.

# 1.1 TỔNG HỢP LÝ THUYẾT

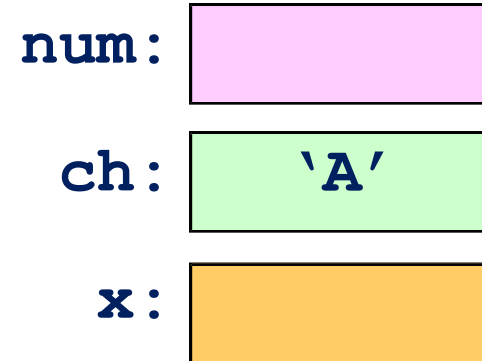
## Các bước sử dụng con trỏ

**Bước 1:** Khai báo một biến sẽ được trỏ bởi biến con trỏ

```
int num;
```

```
char ch = 'A' ;
```

```
float x;
```



# 1.1 TỔNG HỢP LÝ THUYẾT

## Các bước sử dụng con trỏ

### Bước 2: Khai báo con trỏ


```
int num;  
char ch = 'A';  
float x;
```

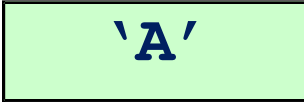
```
int *ptr_num = NULL;  
char *ptr_ch = NULL;  
float *ptr_x = NULL;
```


ptr\_num: NULL

ptr\_ch: NULL

ptr\_x: NULL

num: 

ch: 

x: 

# 1.1 TỔNG HỢP LÝ THUYẾT

## Các bước sử dụng con trỏ

### Bước 3: Tham chiếu con trỏ

```
int num;
```

```
char ch = 'A';
```

```
float x;
```

```
int *ptr_num = NULL;
```

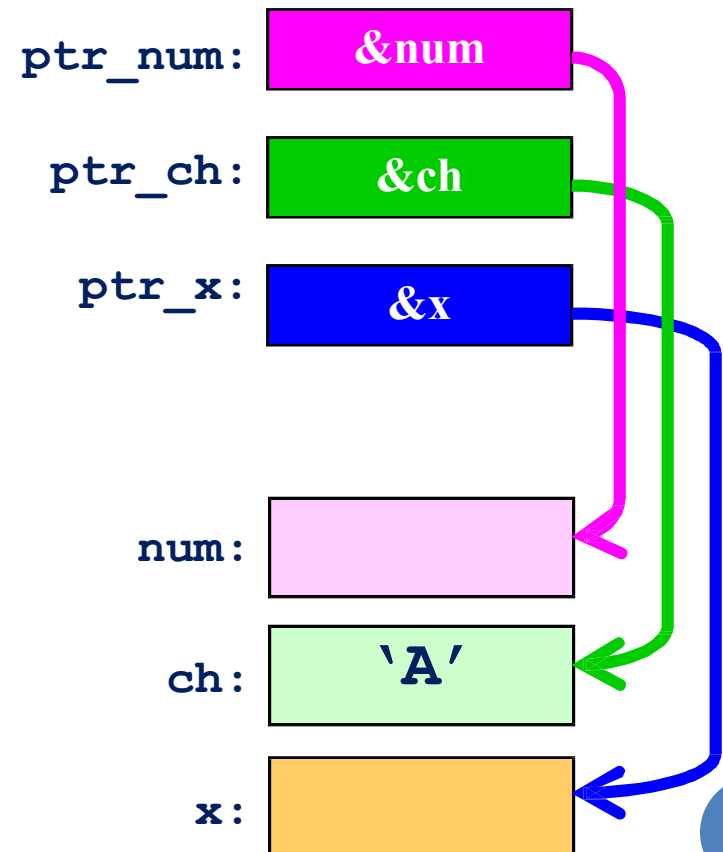
```
char *ptr_ch = NULL;
```

```
float *ptr_x = NULL;
```

```
ptr_num = &num;
```

```
ptr_ch = &ch;
```

```
ptr_x = &x;
```



# 1.1 TỔNG HỢP LÝ THUYẾT

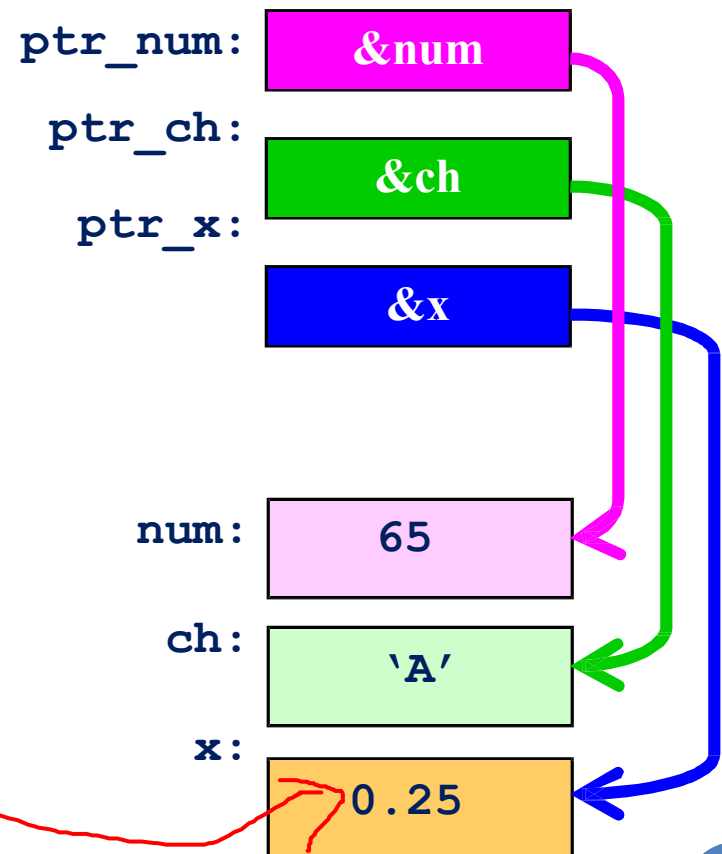


Image Upload

## Các bước sử dụng con trỏ

**Bước 4:** Gán giá trị cho ô nhớ mà con trỏ đang trỏ tới

```
int num;  
char ch = 'A';  
float x;  
int *ptr_num = NULL;  
char *ptr_ch = NULL;  
float *ptr_x = NULL;  
ptr_num = &num;  
ptr_ch = &ch;;  
ptr_x = &x;  
*ptr_x = 0.25;  
*ptr_num = *ptr_ch;
```

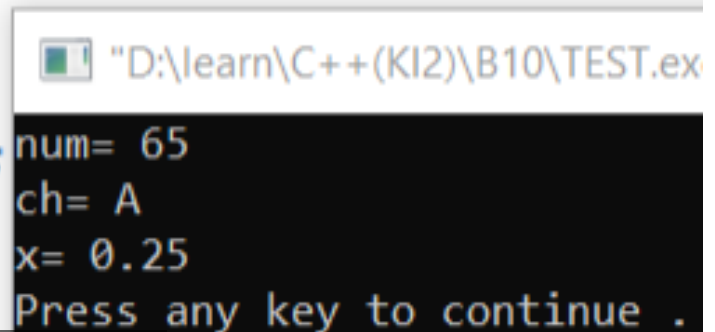


[\*] contro.cpp

```
1  #include<iostream>
2  using namespace std;
3  int main(){
4      int num;
5      char ch = 'A';
6      float x;
7      int *ptr_num=NULL;
8      char *ptr_ch=NULL;
9      float *ptr_x=NULL;
10     ptr_num=&num;
11     ptr_ch=&ch;
12     ptr_x=&x;
13     *ptr_x=0.25;
14     *ptr_x=*ptr_ch;
15     cout << *ptr_ch << endl;
16     cout << *ptr_x << endl;
17     return 0;
18 }
```

*Image uploaded by Nguyễn Sách Nam*

```
1 #include<iostream.h>
2 int main() {
3     int num;
4     char ch = 'A';
5     float x;
6     int *ptr_num=NULL;
7     char *ptr_ch=NULL;
8     float *ptr_x=NULL;
9     ptr_num= &num;
10    ptr_ch= &ch;
11    ptr_x= &x;
12    *ptr_x =0.25;
13    *ptr_num= *ptr_ch;
14    cout<<"num= "<<num<<endl;
15    cout<<"ch= "<<ch<<endl;
16    cout<<"x= "<<x<<endl;
17 }
```



"D:\learn\C++(KI2)\B10\TEST.ex

num= 65  
ch= A  
x= 0.25  
Press any key to continue .

Image uploaded by Nguyễn Tuấn Tiến

# 1.1 TỔNG HỢP LÝ THUYẾT

## Các lỗi thường gặp

Không thể tham chiếu con trỏ đến một hằng số hoặc một phương trình.

- `ptr = &3;`
- `ptr = &(x+5);`

Không thể thay đổi địa chỉ của một biến trong bộ nhớ (vì nó không thể xác định bởi người dùng!)

- `&x = ptr;`
- `&x = 0x2000;`



# 1.1 TỔNG HỢP LÝ THUYẾT

## Biến con trỏ

- Chỉ có thể thực hiện phép toán cộng và trừ trên con trỏ

```
int a, *ptr_a;
```

```
ptr_a = &a;
```

```
a = 500;
```

```
ptr_a ++;
```

- Giả sử biến a được lưu trữ tại địa chỉ 1000
- ptr\_a lưu giá trị 1000. Vì số nguyên có kích thước là 2 bytes, nên sau biểu thức “ptr\_a++;” ptr\_a sẽ có giá trị là 1002 mà không là 1001.

# 1.1 TỔNG HỢP LÝ THUYẾT

**Ví dụ:**

```
int a = 15;           // khai bao bien nguyen
int *p;               // bien con tro p
p = &a;               // cho p tro vao bien a

cout<<"Dia chi cua bien a la: "<<p<<endl;
cout<<"Dia chi cua bien a la: "<<&a<<endl;
cout<<"Gia tri cua bien a la: "<<*p<<endl;
cout<<"Gia tri cua bien a la: "<<a<<endl;
```

# 1.1 TỔNG HỢP LÝ THUYẾT

Hàm new là một trong các hàm được sử dụng thường xuyên nhất để thực hiện việc cấp phát bộ nhớ từ vùng nhớ còn tự do.

**new**      **ten\_kieu\_dl;**

Thực hiện việc cấp phát một vùng nhớ có kích thước bằng kích thước của kiểu dữ liệu.

# 1.1 TỔNG HỢP LÝ THUYẾT

Hàm delete được sử dụng để giải phóng bộ nhớ khi nó không cần dùng nữa.

Cú pháp:

```
delete   ten_con_tro;
```

Ví dụ:            `delete            *ptr;`

Hàm này giải phóng không gian được trả bởi *ptr*, để dùng cho tương lai.

**ptr** phải được dùng trước đó với lời gọi hàm **new**.

# 1.1 TỔNG HỢP LÝ THUYẾT

```
#include <iostream.h>
#include <conio.h>
int main()
{   int n;
    int *ptr;
    cout<<"Nhap so phan tu: ";cin>>n;
    ptr = new    int;           /*cap phat bo nho*/
    if (ptr != NULL)
    {   for(int i = 0 ; i<n ; i++)
        *(ptr+i) = i;
        for(int i=n ; i>0 ; i--)
            cout<<*(ptr+(i-1))<<"\t";
        delete ptr;
        return 0;
    }
    else
    {   cout<<"\nKhong du bo nho de cap phat"<<endl;
        return 1;
    }
}
```

## 1.2 HƯỚNG DẪN THỰC HÀNH BÀI 01



Image Upload

Trong bài thực hành này sinh viên tìm hiểu và thực hành khai báo và sử dụng toán tử &, toán tử \*.

**Bài toán: Viết chương trình thực hiện các công việc sau:**

- Khai báo một biến nguyên và một biến con trỏ
- Lưu trữ địa chỉ của biến nguyên vào biến con trỏ
- Hiển thị giá trị của biến nguyên và in địa chỉ được lưu trữ trong biến con trỏ
- Truy cập giá trị có sẵn tại địa chỉ của biến con trỏ

contro.cpp

```
1 #include<iostream>
2 using namespace std;
3 int main(){
4     int x;
5     cout << "Nhap x: "; cin >> x;
6     int *ptr_x = NULL;
7     ptr_x=&x;
8     cout << "Gia tri cua bien nguyen: " << x << endl;
9     cout << "Dia chi: " << ptr_x << endl;
10    cout << "Gia tri co san cua bien con tro: " << *ptr_x << endl;
11
12    return 0;
13 }
```

E:\Linh tinh\contro.exe

```
Nhap x: 3
Gia tri cua bien nguyen: 3
Dia chi: 0x6ffe04
Gia tri co san cua bien con tro: 3

-----
Process exited after 0.9386 seconds with return value 0
Press any key to continue . . .
```

Image uploaded by Nguyễn Sách Nam

```
1 #include<iostream.h>
2 int main(){
3     int n = 10;
4     int *ptr;
5     ptr = &n;
6     cout<<"n= "<<n<<endl;
7     cout<<"ptr= "<<ptr<<endl;
8     cout<<"*ptr= "<<*ptr<<endl;
9 }
```

"D:\learn\C++(KI2)\B10\TEST.e

```
n= 10
ptr= 0x66ff24
*ptr= 10
Press any key to continue .
```

Image uploaded by Nguyễn Tuấn Tiến



```
co ban (global scope)

#include <iostream>

using namespace std;

int main() {
    int n = 5;
    int* ptr;
    ptr = &n;

    cout << "\n gia tri cua n = " << n; 5
    cout << "\n dia chi cua n : " << &n; ... 84
    cout << "\n gia tri cua ptr = " << *ptr; 5
    cout << "\n dia chi cua ptr : " << ptr; ... 84

    system("pause");
    return 0;
}
```

Image uploaded by nguyên trung hiếu

```
1 #include<iostream>
2 using namespace std;
3 int main() {
4     int n=6;
5     int *num;
6     num = &n;
7     cout<<" n = "<< n << endl;
8     cout<<" Dia chi: "<< &n << endl;
9     cout<<" num = "<< *num << endl;
10    return 0;
11 }
```

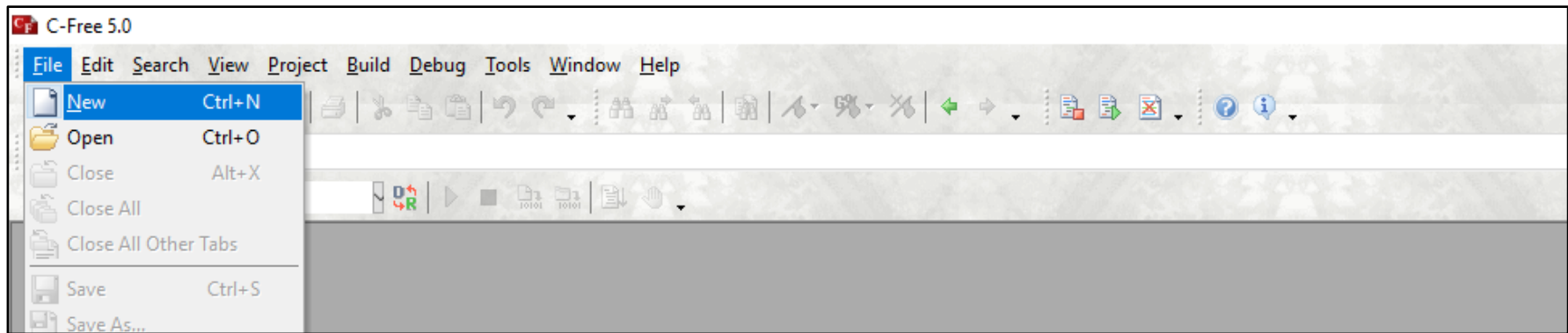
"C:\Users\admin\Documents\C-Free\Temp\Untitled2.exe"

```
n = 6
Dia chi: 0x64ff1c
num = 6
Press any key to continue
```

*Image uploaded by Le Huu Nam*

# 1.2 HƯỚNG DẪN THỰC HÀNH BÀI 01

- **Bước 1:** Tạo mới một file\*.cpp thực hiện thao tác File\New



- File mới xuất hiện, sinh viên chuyển sang bước 2 thực hiện gõ các câu lệnh theo các bước hướng dẫn.

## 1.2 HƯỚNG DẪN THỰC HÀNH BÀI 01

**Bước 2:** Khai báo thư viện cần dùng

```
#include <iostream.h>
```

**Bước 3:** Khai báo hàm main() là hàm chính của chương trình

```
int    main()
```

**Bước 4:** Khai báo biến và lưu trữ địa chỉ của biến con trỏ

```
int a = 15;    // khai bao bien nguyen
```

```
int *p;        // bien con tro p
```

```
p = &a; // luu tru dia chi cua a vao bien  
con tro p
```

## 1.2 HƯỚNG DẪN THỰC HÀNH BÀI 01

**Bước 5:** Hiển thị giá trị của biến nguyên và in địa chỉ được lưu trữ trong biến con trỏ

```
cout << "Gia tri cua bien a la: ";
```

```
cout <<a<< endl;
```

```
//In dia chi duoc luu tru trong bien con tro p
```

```
cout << "Dia chi duoc luu tru trong bien con  
tro p la: ";
```

```
cout <<p<< endl;
```

**Bước 6:** Truy cập giá trị có sẵn tại địa chỉ của biến con trỏ :

```
//Truy cap gia tri co san tai dia chi cua bien con  
tro
```

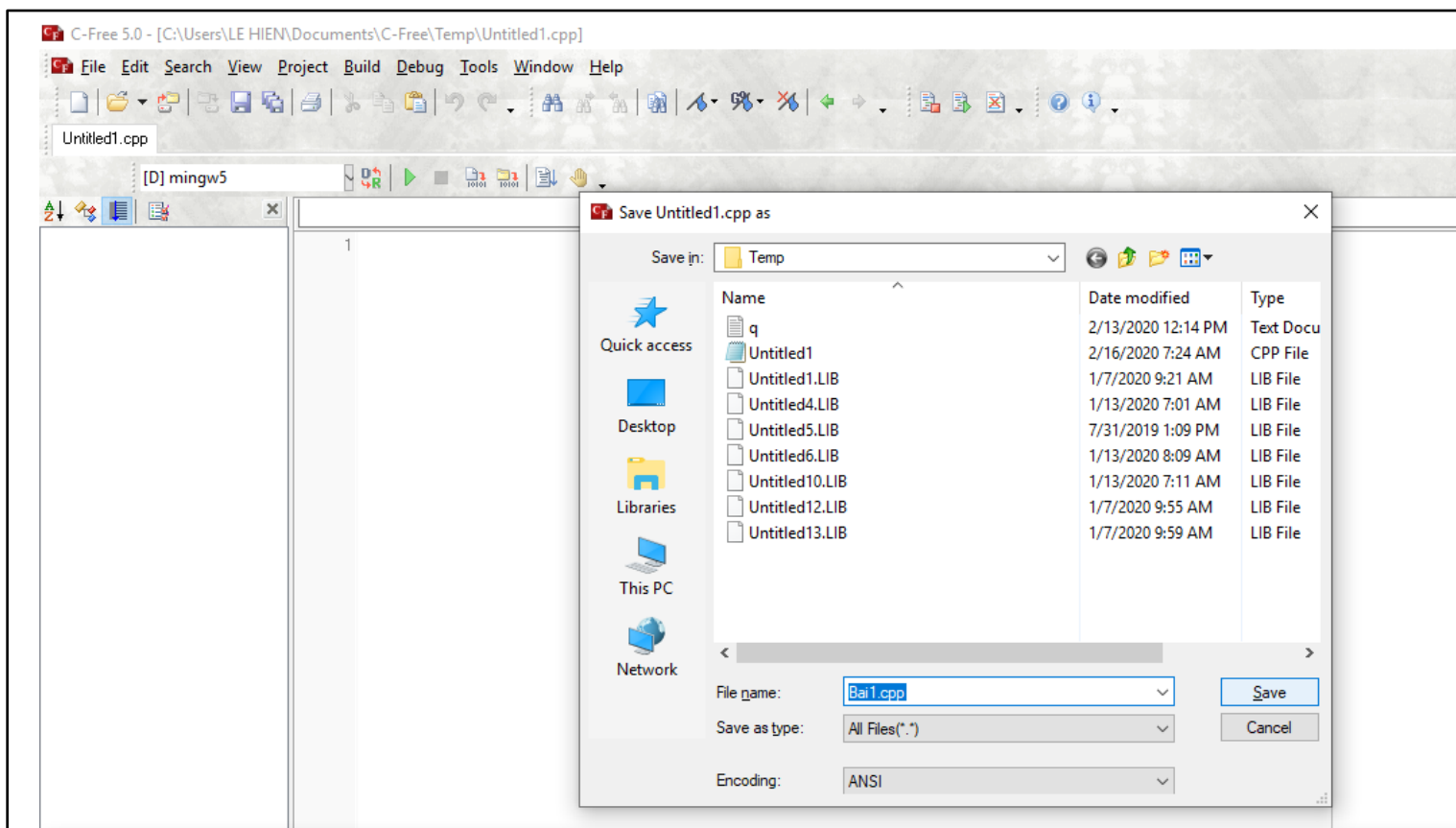
```
cout << "Gia tri cua *p la: ";
```

```
cout << *p<< endl;
```

## 1.2 HƯỚNG DẪN THỰC HÀNH BÀI 01

- **Bước 7:** Lưu bài, chạy kiểm tra chương trình

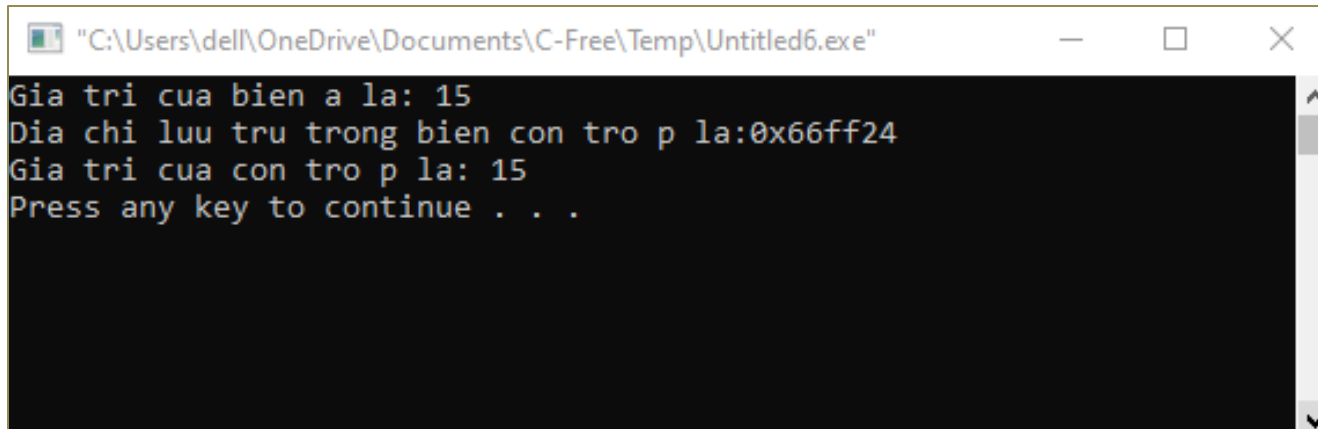
Sinh viên chọn trên thanh công cụ: File\Save\Gõ tên file cần lưu\Save



Ấn F5 để chạy và kiểm tra chương trình

## 1.2 HƯỚNG DẪN THỰC HÀNH BÀI 01

Lưu bài, biên dịch chương trình, nếu có lỗi phải sửa lỗi, nếu chương trình chạy đúng sẽ cho kết quả như sau:



```
"C:\Users\dell\OneDrive\Documents\C-Free\Temp\Untitled6.exe"
Gia tri cua bien a la: 15
Dia chi luu tru trong bien con tro p la: 0x66ff24
Gia tri cua con tro p la: 15
Press any key to continue . . .
```

**Tóm lại:** Trong bài thực hành 01 chúng ta đã biết cách khai báo và sử dụng một con trỏ đơn giản.

## 1.2 HƯỚNG DẪN THỰC HÀNH BÀI 01

```
1. #include <iostream.h>
2. int    main()
3. {   int a = 15;           // khai bao bien nguyen
4.     int *p;               // bien con tro p
5.     p = &a;
6.     //luu tru dia chi cua a vao bien con tro p
7.     cout << "Gia tri cua bien a la: "<<a<< endl;
8.     //In dia chi duoc luu tru trong bien con tro p
9.     cout<<"Dia chi luu tru trong bien con tro p la:";
10.    cout<<p<< endl;
11.    cout<<"Gia tri cua con tro p la: "<<*p<< endl;
12.    return 0;
13. }
```



## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

Trong bài thực hành này sinh viên tiếp tục tìm hiểu và thực hành cách sử dụng con trỏ.



Image Upload

### Bài toán:

Viết chương trình thực hiện các công việc sau:

- Khai báo 2 biến nguyên  $i$ ,  $j$  và 2 biến con trỏ  $p$ ,  $q$ , sau đó cho 2 biến con trỏ này lần lượt trỏ đến 2 biến nguyên  $i$  và  $j$  vừa khai báo.
- In ra địa chỉ của biến nguyên  $i$ ; in ra địa chỉ biến nguyên  $j$  thông qua biến con trỏ trỏ tới biến  $j$ .
- Gán cho biến  $i$  một giá trị, gán cho biến  $j$  một giá trị thông qua biến con trỏ trỏ tới biến  $j$
- Thực hiện tăng giá trị của biến  $i$  và  $j$  thông qua các biến con trỏ trỏ tới chúng và in giá trị của 2 biến  $j$  và  $i$  vừa tăng ra màn hình.

## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

**Bước 1:** Tạo file mới: tương tự như hướng dẫn ở bài thực hành 01

**Bước 2:** Khai báo thư viện cần dùng

```
#include <iostream.h>
```

```
#include <conio.h>
```

## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

**Bước 3:** Khai báo 2 biến nguyên i, j và 2 biến con trỏ p, q, sau đó cho 2 biến con trỏ này lần lượt trỏ đến 2 biến nguyên i và j vừa khai báo

```
int i, j ; //khai báo 2 biến nguyên i, j
int *p, *q ; //khai báo 2 con trỏ nguyên p, q
p = &i;    //cho p trỏ tới i
q = &j;    //cho q trỏ tới j
```

## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

**Bước 4:** In ra địa chỉ của biến nguyên i; in ra địa chỉ biến nguyên j thông qua biến con trỏ trỏ tới biến j.

```
// hỏi địa chỉ biến i
```

```
cout << "Địa chỉ biến i là: " << &i << endl;
```

```
// hỏi địa chỉ biến j (thông qua q)
```

```
cout << "Địa chỉ biến j là: " << q << endl ;
```

## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

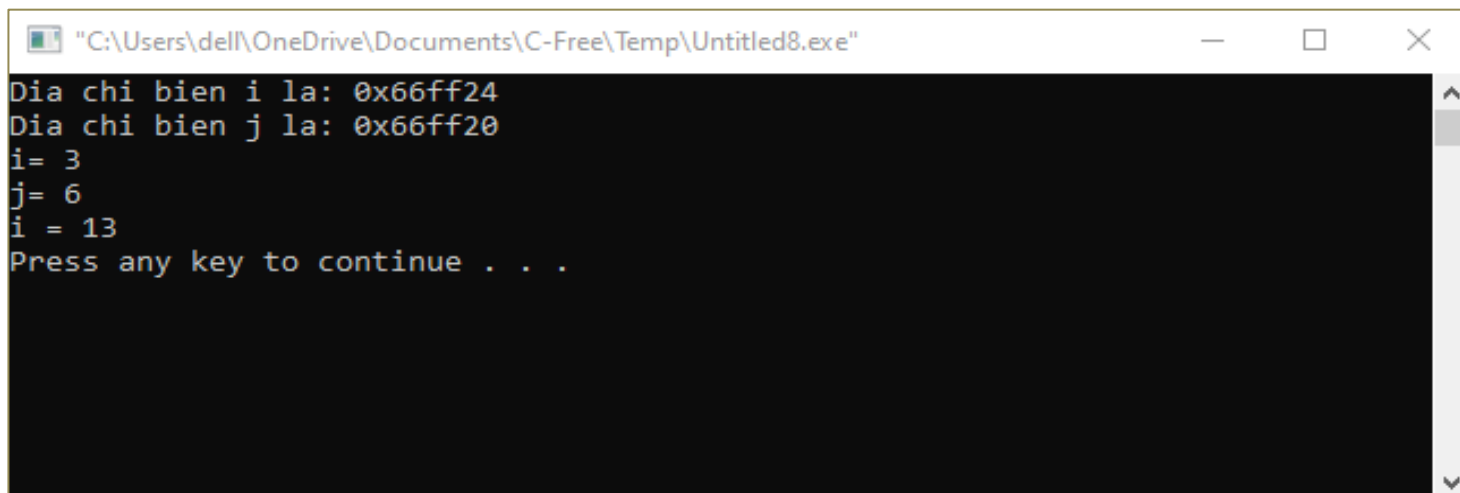
**Bước 5:** Gán cho biến i một giá trị, gán cho biến j một giá trị thông qua biến con trỏ trỏ tới biến j. Thực hiện tăng giá trị của biến i và j thông qua các biến con trỏ trỏ tới chúng và in giá trị của 2 biến j và i vừa tăng ra màn hình.

```
i = 2;                //gán i bằng 2
*q = 5;              //gán j bằng 5 (thông qua q)
i=i+1;
cout<<"i= " <<i<<endl; //tăng i và hỏi i, i = 3
(*q)++;
cout<<"j= " <<j<<endl; //tăng j (thông qua q)
và hỏi j, j = 6
(*p) = (*q) * 2 + 1; //gán lại i (thông qua p)
cout <<"i = " << i <<endl; // 13
```

## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

**Bước 6:** Thực hiện tương tự như bài thực hành 01

Biên dịch chương trình, nếu có lỗi phải sửa lỗi, nếu chương trình chạy đúng sẽ cho kết quả như sau:



```
"C:\Users\del\OneDrive\Documents\C-Free\Temp\Untitled8.exe"
Dia chi bien i la: 0x66ff24
Dia chi bien j la: 0x66ff20
i= 3
j= 6
i = 13
Press any key to continue . . .
```

**Tóm lại:** Trong bài thực hành này chúng ta sẽ biết cách sử dụng câu lệnh toán tử \* và toán tử &.

## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

```
1.#include <iostream.h>
2.int main()
3.{ int i, j ; //khai báo 2 biến nguyên i, j
4.  int *p, *q ; //kb 2 con trỏ nguyên p,q
5.  p = &i;      //cho p trỏ tới i
6.  q = &j;      //cho q trỏ tới j
7.  cout <<"Địa chỉ biến i là: "<<&i<<endl;
8.  cout <<"Địa chỉ biến j là: "<<q<<endl ;
9.  i = 2;       //gán i bằng 2
10. *q = 5;      //gán j bằng 5 (thông qua q)
```

## 1.3 HƯỚNG DẪN THỰC HÀNH BÀI 02

```
11.    i++;  
12.    cout<<"i= " <<i<<endl; //tăng i và in i = 3  
13.    (*q)++ ;  
14.    cout<<"j= " <<j<<endl; //tăng j và in j = 6  
15.    (*p) = (*q) * 2 + 1; //gán lại i thông qua p  
16.    cout <<"i = " << i <<endl; // 13  
17.    return 0;  
18. }
```



## II. HƯỚNG DẪN THƯỜNG XUYỀN

- Sinh viên tiếp tục thực hành các bài tập theo sự gợi ý hướng dẫn.

Trong phần này yêu cầu:

- + Một số phần trong bài sinh viên phải tự thực hiện (Ví dụ: tạo file ban đầu, khai báo thư viện, những yêu cầu sử dụng con trỏ có tính chất tương tự ở ví dụ trước).
- + Sinh viên phải tự hoàn thiện chương trình và chạy đúng.

## 2.1 HƯỚNG DẪN THỰC HÀNH BÀI 03

**Bài toán:** Viết một chương trình sử dụng biến con trỏ để cộng hai số nguyên a và b.

**Đoạn lệnh sử dụng con trỏ để tính tổng 2 số nguyên**

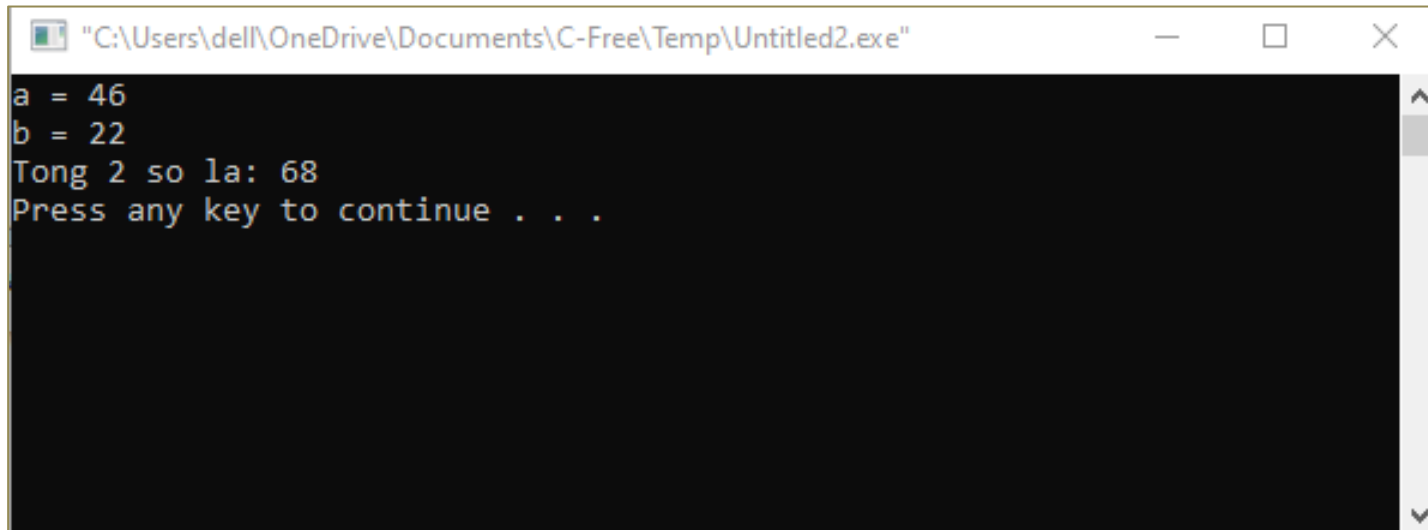
```
int *ptr1, *ptr2;  
ptr1 = &a;    //con trỏ ptr1 trỏ vào biến a  
ptr2 = &b;    //con trỏ ptr2 trỏ vào biến b  
int sum;      //biến sum để chứa tổng  
sum = *ptr1 + *ptr2;
```

## 2.1 HƯỚNG DẪN THỰC HÀNH BÀI 03

```
1. #include <iostream.h>
2. int main()
3. { int a, b, *ptr1, *ptr2;
4.   cout<<"a = ";cin>>a;
5.   cout<<"b = ";cin>>b;
6.   ptr1 = &a;   //con tro ptr1 tro vao bien a
7.   ptr2 = &b;   //con tro ptr2 tro vao bien b
8.   int sum;     //bien sum de chua tong
9.   sum = *ptr1 + *ptr2;
10.  cout<<"Tong 2 so la: "<<sum<<endl;
11.  return 0;
12. }
```

## 2.1 HƯỚNG DẪN THỰC HÀNH BÀI 03

- Chương trình hoàn chỉnh sinh viên xem trong tài liệu “B10\_Tailieu\_TTLTCB\_1”
- Kết quả khi chạy chương trình



```
"C:\Users\dell\OneDrive\Documents\C-Free\Temp\Untitled2.exe"
a = 46
b = 22
Tong 2 so la: 68
Press any key to continue . . .
```

## 2.2 HƯỚNG DẪN THỰC HÀNH BÀI 04

**Bài toán:** Viết chương trình hoán vị hai số A và B có sử dụng biến con trỏ.

**Hàm hoán vị**

```
void hoanvi(float *x, float *y)
{
    float  tg;
    tg = *x;
    *x = *y;
    *y = tg;
}
```

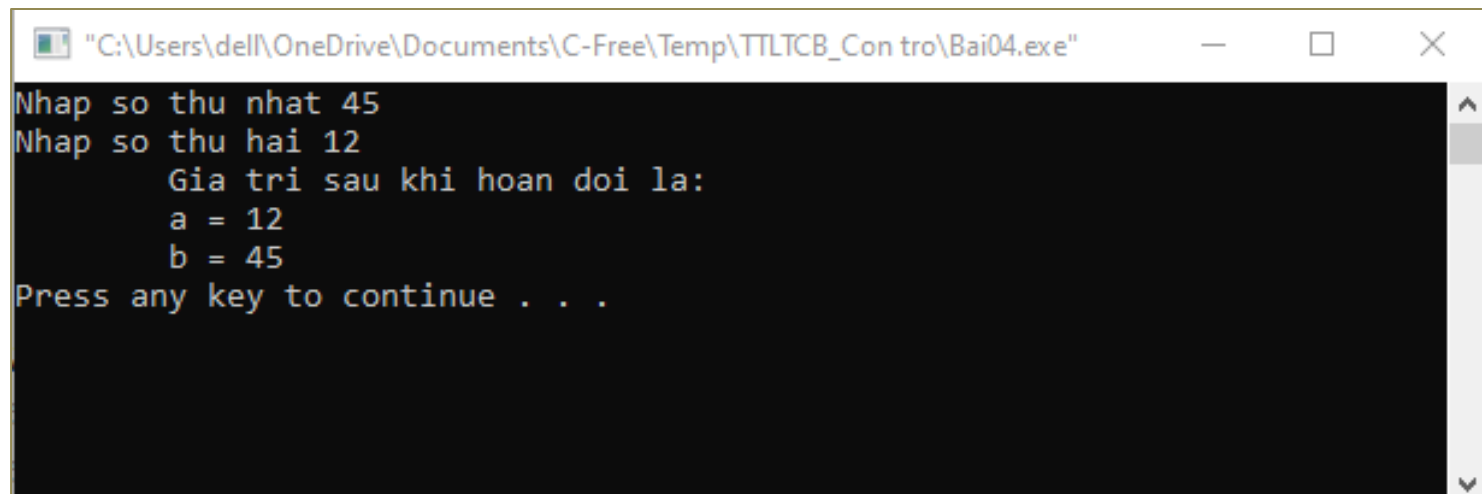
**Gọi hàm này vào trong main()**

## 2.2 HƯỚNG DẪN THỰC HÀNH BÀI 04

```
#include<iostream.h>
void hoanvi (float *x, float *y)
{
    float tg = *x;
    *x = *y;
    *y = tg;
}
int main ()
{
    float    a, b;
    cout<<"Nhap so thu nhat ";cin>>a;
    cout<<"Nhap so thu hai ";cin>>b;
    hoanvi (&a, &b) ;
    cout<<"Gia tri sau hoan doi la"<<endl;
    cout<<"a = " <<a<<" b = " <<b<<endl;
}
```

## 2.2 HƯỚNG DẪN THỰC HÀNH BÀI 04

- Chương trình hoàn chỉnh sinh viên xem trong tài liệu “B10\_Tailieu\_TTLTCB\_1”
- Kết quả khi chạy chương trình



```
"C:\Users\dell\OneDrive\Documents\C-Free\Temp\TTLTCB_Con tro\Bai04.exe"
Nhap so thu nhat 45
Nhap so thu hai 12
    Gia tri sau khi hoan doi la:
    a = 12
    b = 45
Press any key to continue . . .
```

### III. BÀI TẬP TỰ GIẢI

**Bài 5.** Hãy viết đoạn chương trình để khai báo biến số nguyên a và xuất ra địa chỉ ô nhớ được cấp phát cho biến này khi chương trình chạy.

**Bài 6.** Sử dụng biến con trỏ viết chương trình nhập vào 3 số nguyên a,b,c và tìm số lớn nhất trong 3 số nguyên vừa nhập.

**Bài 7.** Viết chương trình để giải phương trình bậc nhất có sử dụng biến con trỏ



### III. BÀI TẬP TỰ GIẢI

**Bài 8.** Hãy khai báo biến con trỏ p các số nguyên và viết lệnh cấp phát không gian bộ nhớ để biến con trỏ p có thể chứa 100 số nguyên. Sau đó viết lệnh thu hồi lại không gian bộ nhớ đã cấp phát cho con trỏ.

**Bài 9:** Viết chương trình tính giai thừa của một số có sử dụng biến con trỏ.

**Bài 10.** Viết chương trình sử dụng biến con trỏ để giải phương trình bậc 2:  $ax^2 + bx + c = 0$  ( $a \neq 0$ ) với a, b, c nhập vào từ bàn phím.

## IV. TỔNG KẾT

**Kiến thức cần ghi nhớ trong buổi thực hành:**

**1** Khái niệm con trỏ, Cách khai báo con trỏ

**2** Sử dụng 2 toán tử  $&$ ,  $*$

**3** Cấp phát động

**4** Phát hiện và sửa lỗi

## V. GIAO NHIỆM VỤ TUẦN TIẾP THEO

1. Hoàn thành tất cả các bài thực hành trong buổi học và bài tập trong mục D file “B10\_Baitap\_TTLTCB\_1.docx”.
2. Sinh viên đọc tài liệu và chuẩn bị trước nội dung học của tuần tiếp theo:
  - **Lập trình với con trỏ phần 2**
    - + Khai báo mảng con trỏ, xâu con trỏ
    - + Các thao tác trên mảng con trỏ, xâu con trỏ
    - + Viết chương trình sử dụng mảng/xâu con trỏ
    - + Phát hiện và sửa lỗi.