



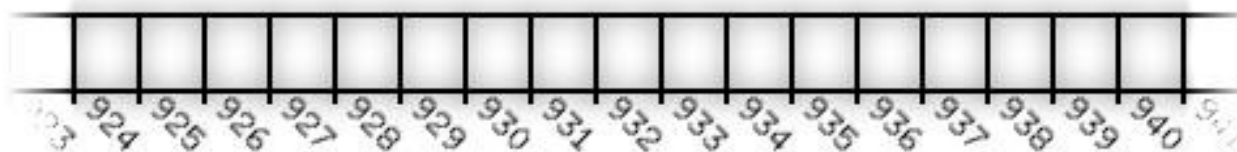
ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

C PROGRAMMING INTRODUCTION

TUẦN 11: CON TRỎ[?]

Địa chỉ bộ nhớ

- Bộ nhớ máy tính bao gồm các byte. Mỗi byte có một **địa chỉ** đi kèm với nó
- VD : địa chỉ từ 924 tới 940



Địa chỉ bộ nhớ

- Toán tử **&** cho địa chỉ của một biến

```
#include <stdio.h>
```

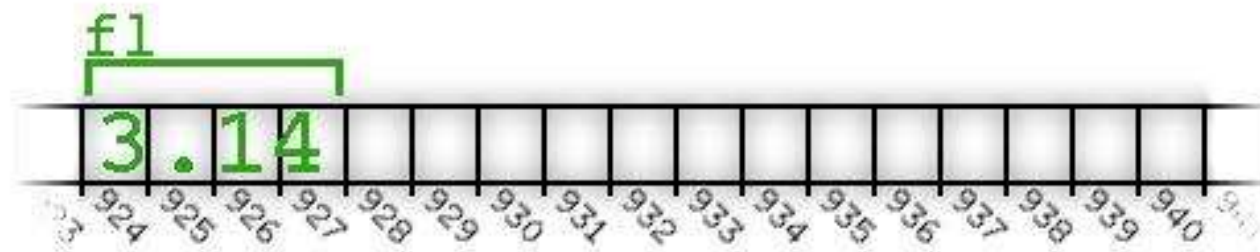
```
int main() {
```

```
float f1=3.14;
```

```
printf("f1's address=%u\n", (unsigned int) &f1);
```

```
return 0;
```

```
}
```



Bài tập 11.1

- Viết chương trình C sử dụng 3 biến để nhập 3 số nguyên. Với mỗi biến, sử dụng một con trỏ để chỉ đến các biến này sau đó hiển thị giá trị của biến được trỏ đến bởi con trỏ.

Lời giải

```
#include <stdio.h>

int main(){
    int x, y, z;
    int* ptr;
    printf("Enter three integers: ");
    scanf("%d %d %d", &x, &y, &z);
    printf("\nThe three integers are:\n");
    ptr = &x;
    printf("x = %d\n", *ptr);
    ptr = &y;
    printf("y = %d\n", *ptr);
    ptr = &z;
    printf("z = %d\n", *ptr);
    return 0;
}
```

Bài tập 11.2

- Viết chương trình in ra địa chỉ của 5 phần tử đầu tiên của mảng dưới đây:

`int a[7]= {13, -355, 235, 47, 67, 943, 1222} ;`

Lời giải

```
#include <stdio.h>

int main() {
    int a[7]= {13, -355, 235, 47, 67, 943,
1222};

    int i;
    printf("address of first five elements in
memory.\n");
    for (i=0; i<5;i++)printf("\ta[%d]",i);
    printf("\n");
    for (i=0; i<5;i++)printf("\t%p",&a[i]);
    return 0;
}
```

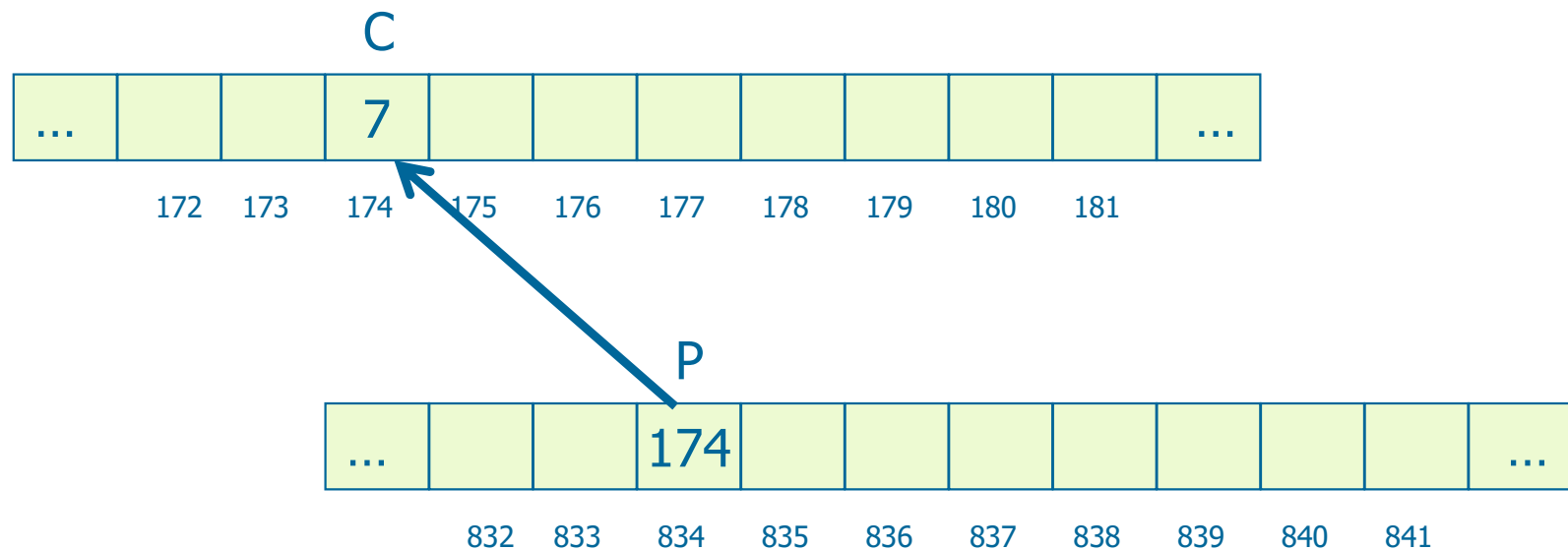
Khai báo một biến con trỏ

```
type *variable_name;
```

- Con trỏ được khai báo bằng cách thêm * trước tên biến.
- Con trỏ là một biến có giá trị là một địa chỉ trong bộ nhớ.
- Địa chỉ của biến hoặc mảng được khai báo.

Con trỏ

- *ptr* trỏ đến địa chỉ của biến *c*

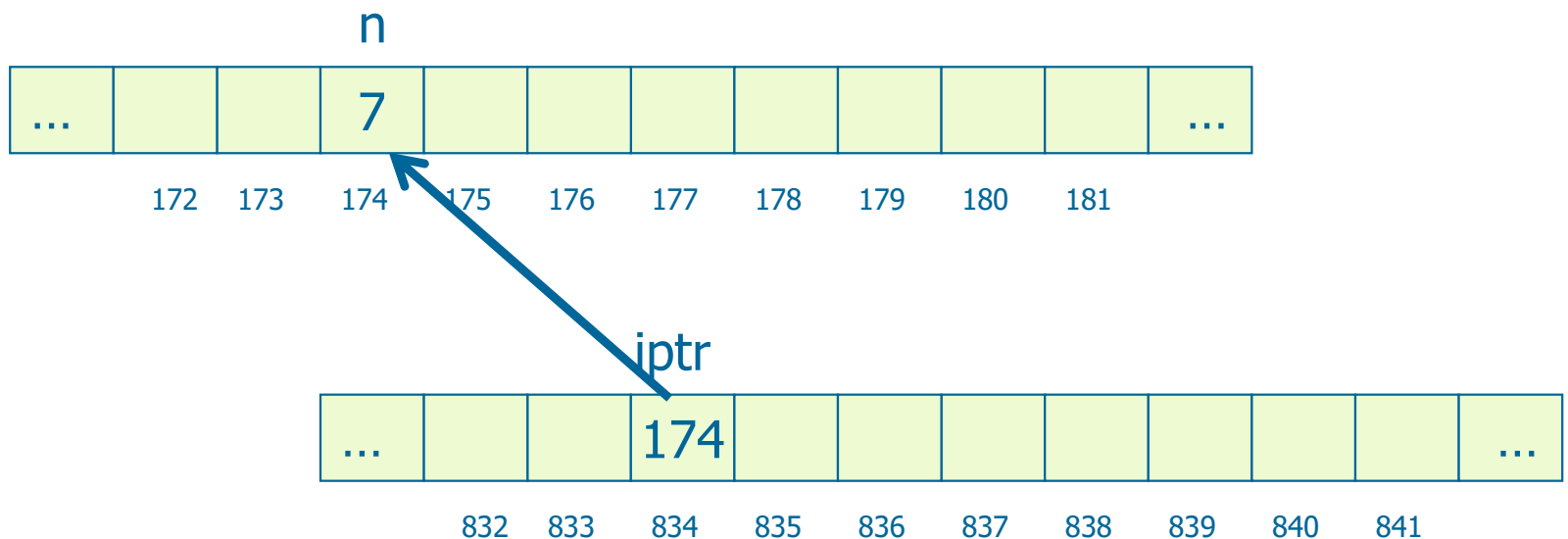


Tham chiếu

- Toán tử **&** cho địa chỉ của biến
- Câu lệnh: **ptr = &c;**
- gán địa chỉ của **c** tới biến con trỏ **ptr**, và **ptr** trỏ tới **c**
- Để in một con trỏ, sử dụng định dạng **%p**.

Tham chiếu

```
int n;  
int *iptr; /* Declare P as a pointer to int */  
n = 7;  
iptr = &n;
```



Khử tham chiếu

- Toán tử $*$ dùng để khử tham chiếu
- Áp dụng trên các con trỏ
- Truy cập tới đối tượng mà con trỏ trỏ tới
- Câu lệnh: $*iptr = 5;$
gán vào n (biến mà $iptr$ trỏ tới) giá trị 5

Bài tập 11.3

- Viết chương trình yêu cầu người dùng nhập vào 3 số thực a, b, c. Thêm 100 vào 3 số chỉ sử dụng một con trỏ

Lời giải

```
#include <stdio.h>
void main(void)
{
    int x = 25, y = 50, z = 75;
    int *ptr;
    printf("Here are the values of x, y, and z:\n");
    printf("%d %d %d\n", x, y, z);
    ptr = &x; // Store the address of x in ptr
    *ptr += 100; // Add 100 to the value in x
    ptr = &y; // Store the address of y in ptr
    *ptr += 100; // Add 100 to the value in y
    ptr = &z; // Store the address of z in ptr
    *ptr += 100; // Add 100 to the value in z
    printf("Once again, here are the values of x, y,
    and z:\n");
    printf("%d %d %d\n", x, y, z);
}
```

Truyền tham số bởi giá trị

- Hàm nhận tham số bởi giá trị của bản sao của biến của hàm gọi tới
- Giá trị truyền vào có thể được thay đổi trong hàm
- Giá trị của biến ở hàm gọi tới không thể được thay đổi

Hàm swap lỗi

Hàm swap nhận giá trị các biến và **không** thay đổi giá trị của các biến gốc

```
void swap(int x, int y) {  
    int tmp = x;  
    x = y;  
    y = tmp;  
}
```


Sửa lỗi

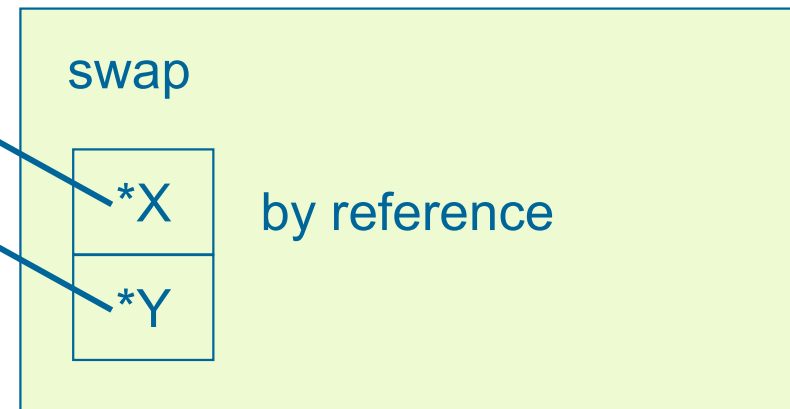
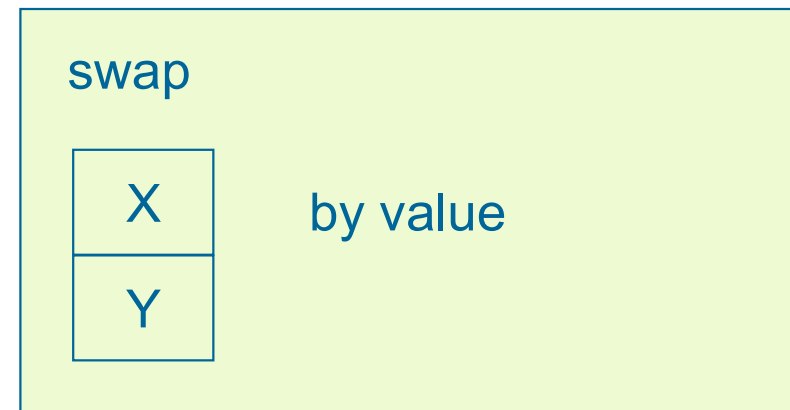
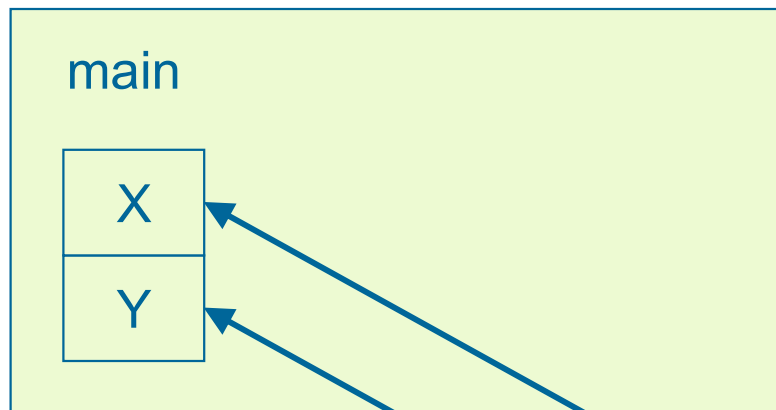
- Cho hàm swap nhận con trỏ tới các số nguyên

```
void swap(int *x, int *y)
{
    int temp = *x;
    *x = *y;
    *y = temp;
}
```

- Gọi hàm `swap(&x, &y);`
- Truyền tham chiếu

Caller

Called



Bài tập 11.4

- Viết hàm nhận 3 biến a, b, c và đổi chỗ sao cho giá trị của a thành b , b thành c và c thành a . Kiểm tra hàm này bằng một chương trình

Lời giải

```
#include <stdio.h>
void swap3(int *p, int *q, int *r){
    int tmp;
    tmp= *p; *p=*q; *q=*r; *r=tmp;
}
void main(void)
{
    int a, b, c;
    printf("Enter a, b, c:");
    scanf("%d%d%d", &a, &b, &c);
    printf("Value before swap. a=%d, b=%d, c=%d\n", a,
b, c);
    swap3(&a, &b, &c);
    printf("Value after swap. a=%d, b=%d, c=%d\n", a,
b, c);
}
```

Bài tập 11.5

Khai báo các biến số nguyên x, y, z và các con trỏ số nguyên p, q, r . Khởi tạo giá trị cho x, y, z . Khởi tạo p, q, r tới địa chỉ của x, y, z .

- 1) In ra giá trị của $x, y, z, p, q, r, *p, *q, *r$.
- 2) Đổi chỗ các giá trị x, y, z . In ra giá trị của $x, y, z, p, q, r, *p, *q, *r$.
- 3) Đổi chỗ giá trị p, q, r . In ra giá trị của $x, y, z, p, q, r, *p, *q, *r$.

Bài tập 11.6

- Viết hàm *incomeplus* để nâng lương 300000 cho nhân viên với điều kiện số năm làm việc > 3

`void incomeplus(long *current, int year)`

- Kiểm tra hàm với một chương trình

Lời giải

```
#include <stdio.h>
void incomeplus(long *current, int year){
    if (year >3) *current = *current + 300000;
}
void main(void)
{
    long cursal; int year;
    do {
        printf("Enter your current salary:");
        scanf("%ld",&cursal);
        printf("Number of years passed:");
        scanf("%d", &year);
        incomeplus(&cursal,year);
        printf("Your salary now: %ld", cursal);
    }while(year!=-1);
```



25 YEARS ANNIVERSARY
SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

**Thank you
for your
attentions!**



soict.hust.edu.vn/



fb.com/groups/soict

