



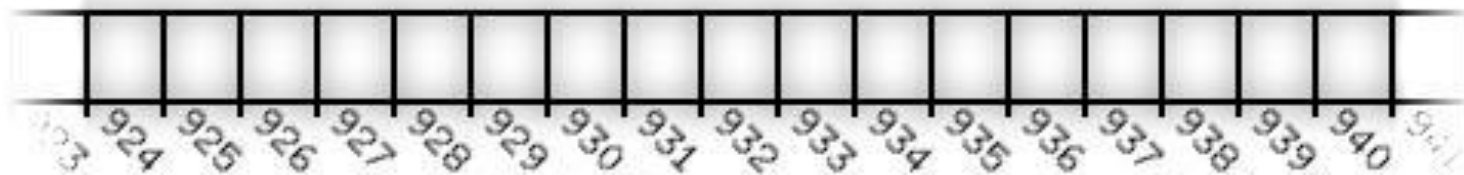
ĐẠI HỌC BÁCH KHOA HÀ NỘI  
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

# 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

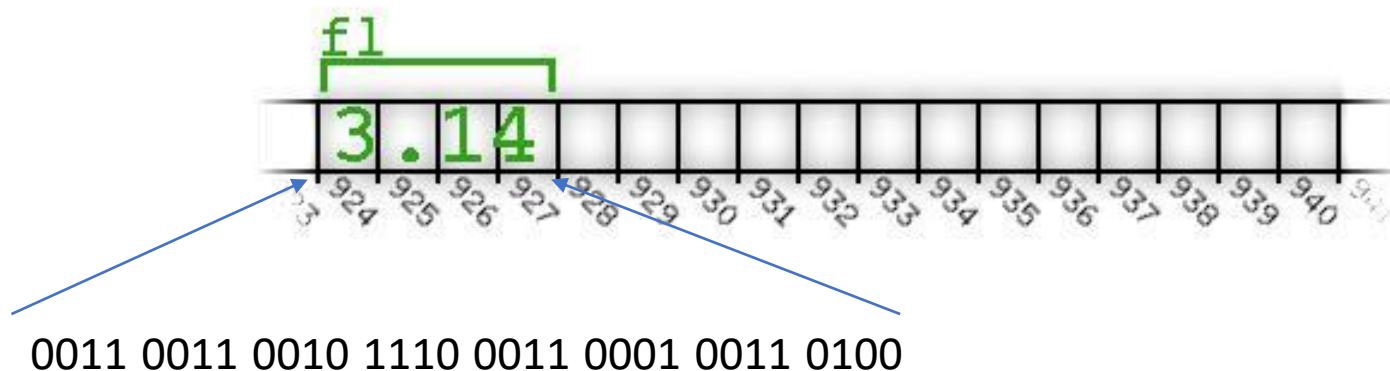


# Địa chỉ bộ nhớ

## ■ 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;
}
```



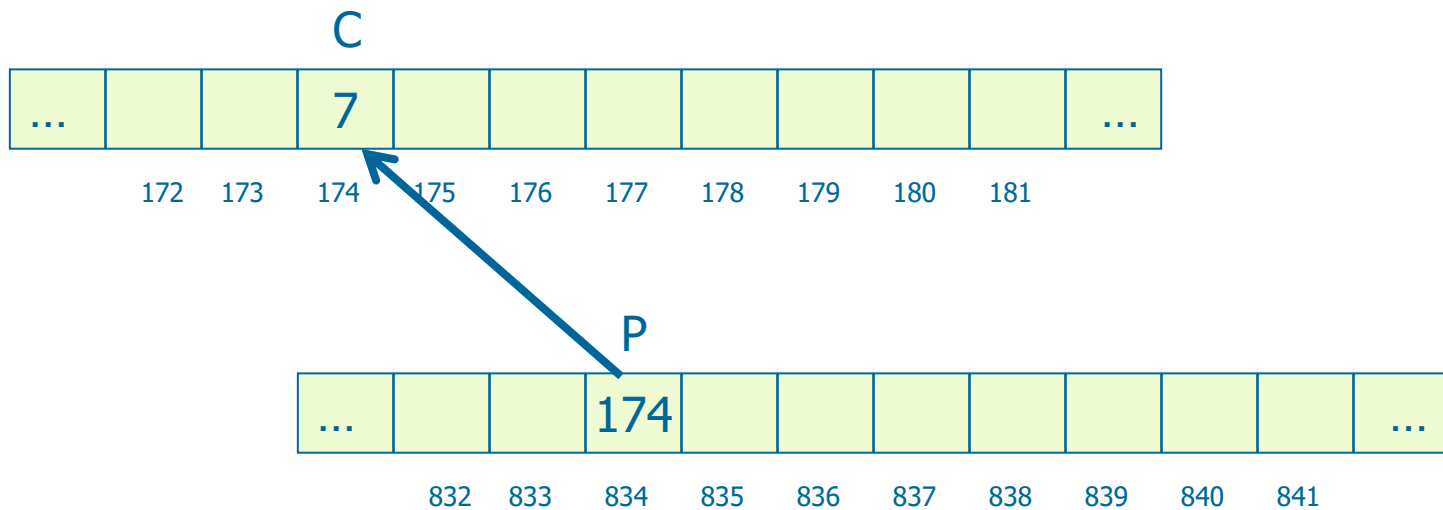
# Khai báo một biến con trỏ

```
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**

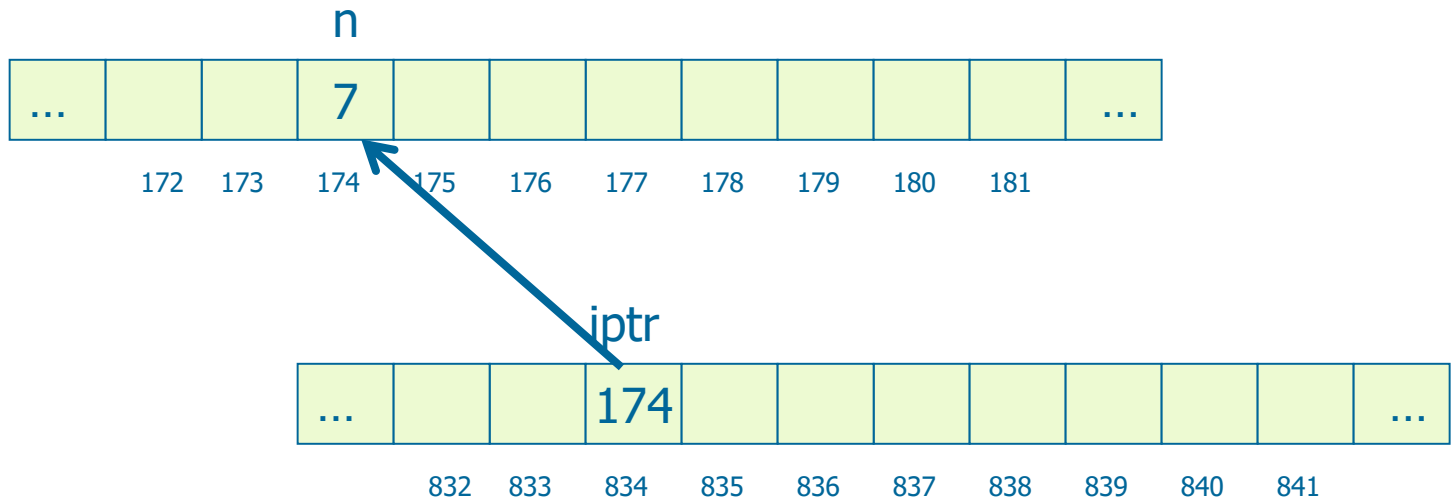


# Tham chiếu

- 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 11.1

- 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.

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

# Lời giải

```
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;
}
```

# 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;  
}
```



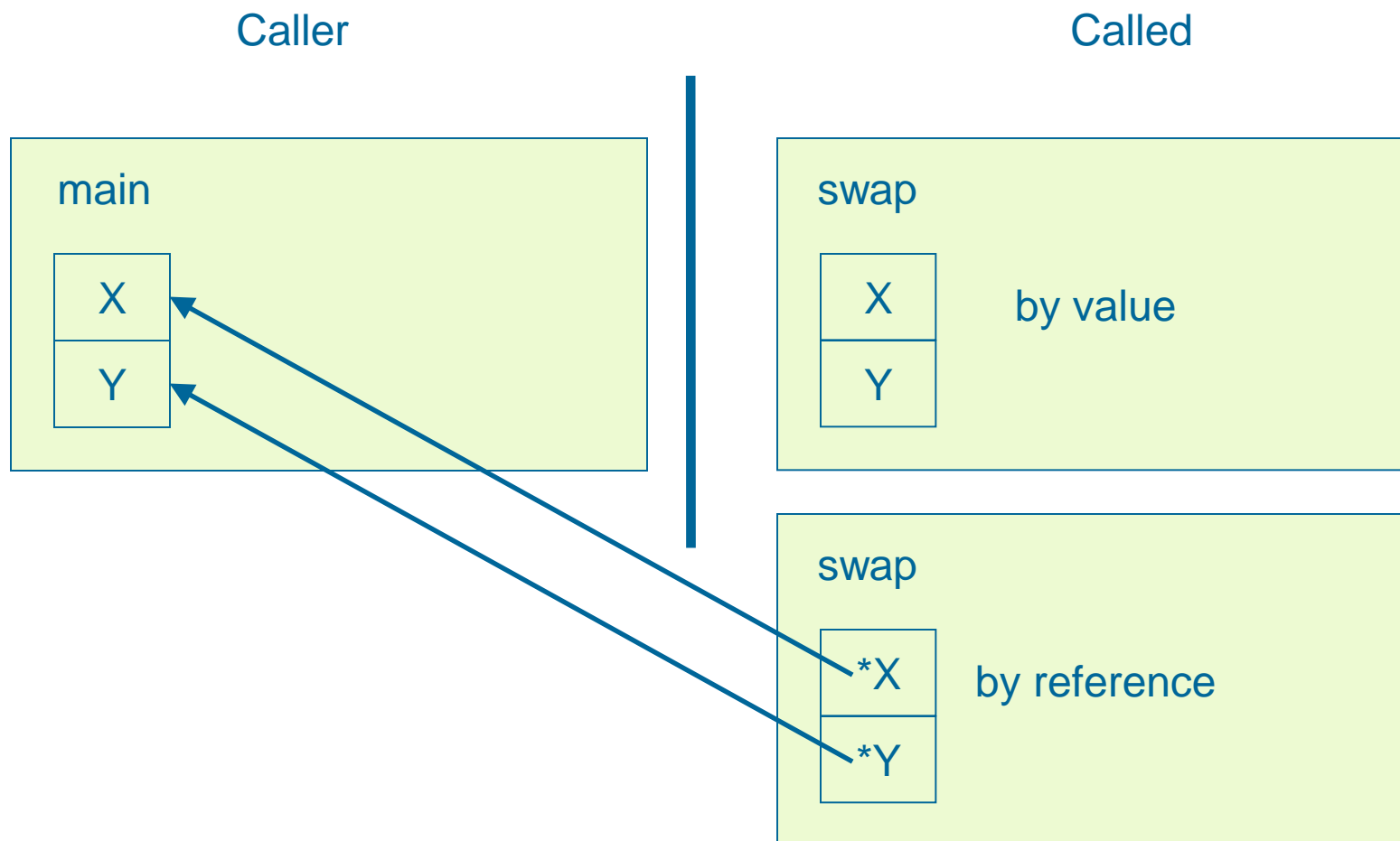
# Sửa lỗi

- 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

# Truyền tham số bởi con trỏ



# 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);
}
```

# Bài tập 11.7

1. Write a program in C to show the basic declaration of pointer.

Expected Output :

Pointer : Show the basic declaration of pointer :

-----  
Here is m=10, n and o are two integer variable and \*z is an integer

z stores the address of m = 0x7ffd40630d44

\*z stores the value of m = 10

&m is the address of m = 0x7ffd40630d44

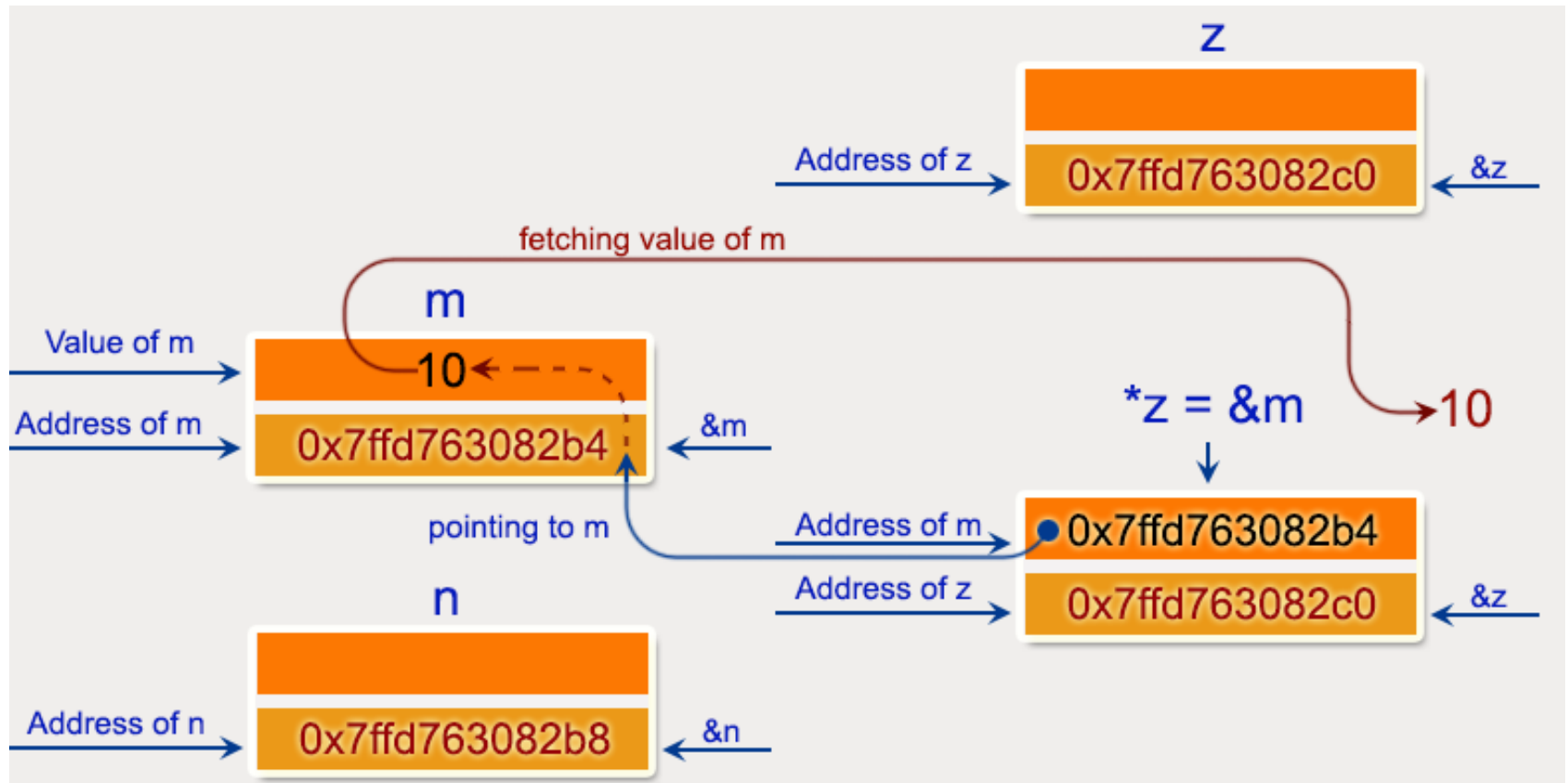
&n stores the address of n = 0x7ffd40630d48

&o stores the address of o = 0x7ffd40630d4c

&z stores the address of z = 0x7ffd40630d50



# Bài tập 11.7



# Bài tập 11.8

2. Write a program in C to demonstrate how to handle the pointers in the program. Go to the editor

Expected Output :

Address of m : 0x7ffcc3ad291c

Value of m : 29

Now ab is assigned with the address of m.

Address of pointer ab : 0x7ffcc3ad291c

Content of pointer ab : 29

The value of m assigned to 34 now.

Address of pointer ab : 0x7ffcc3ad291c

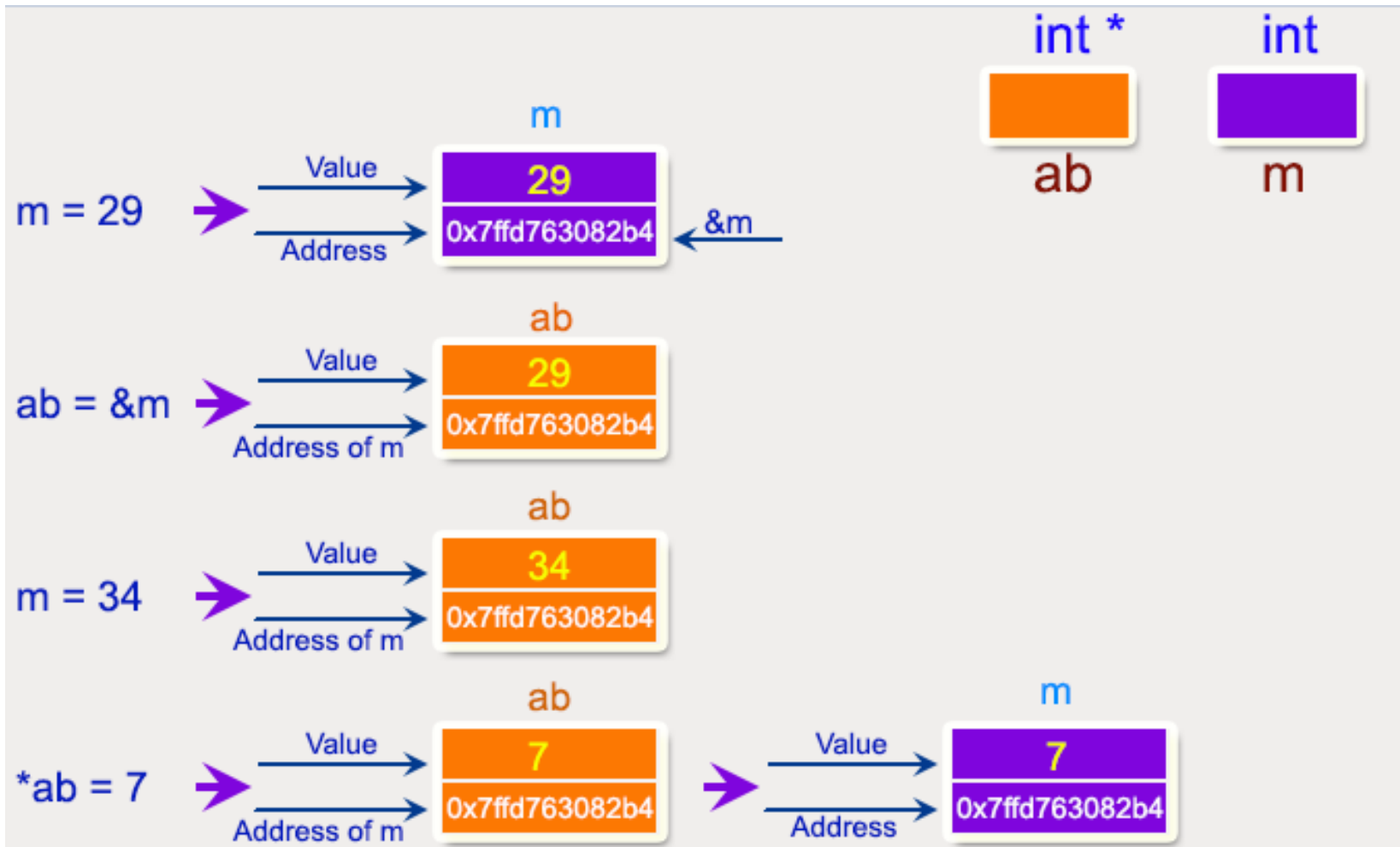
Content of pointer ab : 34

The pointer variable ab is assigned with the value 7 now.

Address of m : 0x7ffcc3ad291c

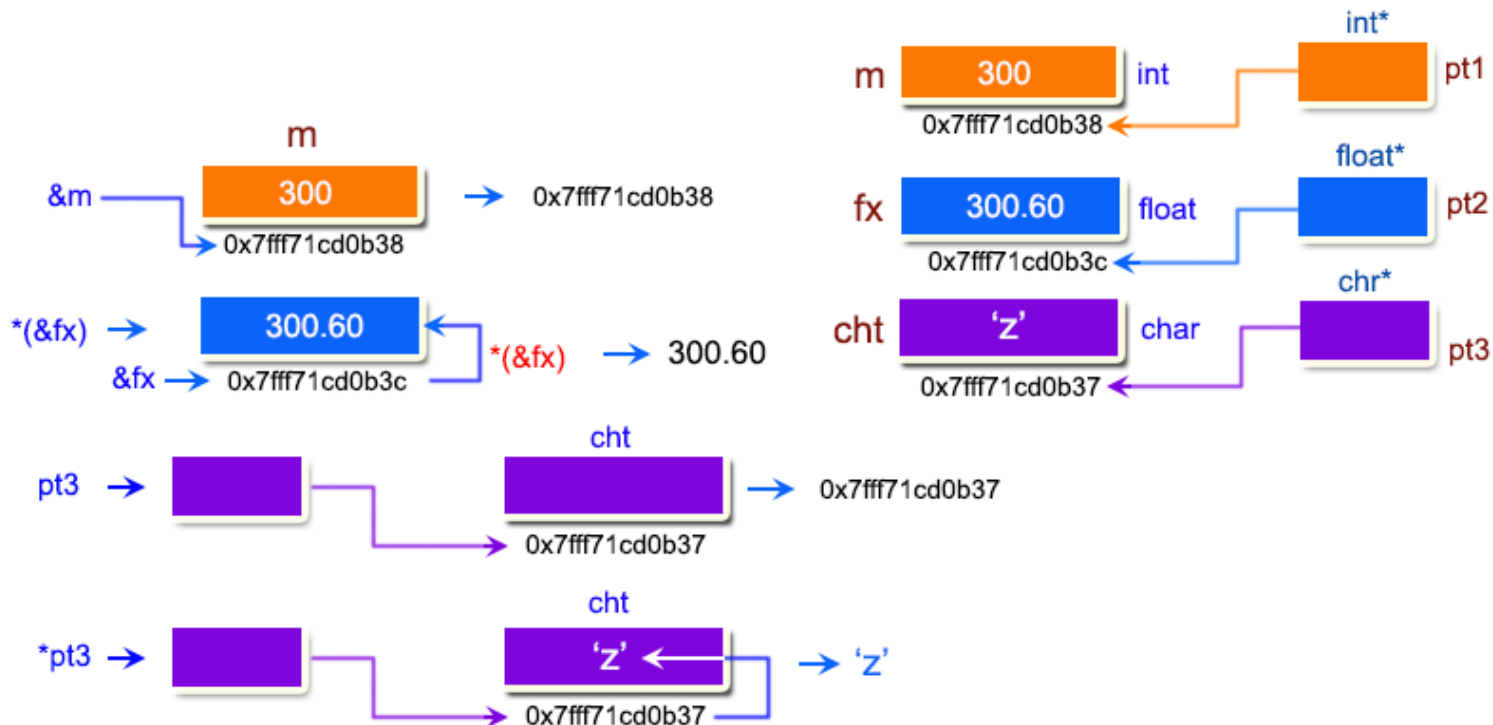
Value of m : 7

# Bài tập 11.8



# Bài tập 11.9

- Write a program in C to demonstrate the use of `&`(address of) and `*`(value at address) operator.



# Bài tập 11.9

Expected Output :

Pointer : Demonstrate the use of & and \* operator :

-----  
m = 300

fx = 300.600006

cht = z

Using & operator :

-----  
address of m = 0x7ffda2eeeeec8

address of fx = 0x7ffda2eeeecc

address of cht = 0x7ffda2eeeeec7

Using & and \* operator :

-----  
value at address of m = 300

value at address of fx = 300.600006

value at address of cht = z

Using only pointer variable :

-----  
address of m = 0x7ffda2eeeeec8

address of fx = 0x7ffda2eeeecc

address of cht = 0x7ffda2eeeeec7

Using only pointer operator :

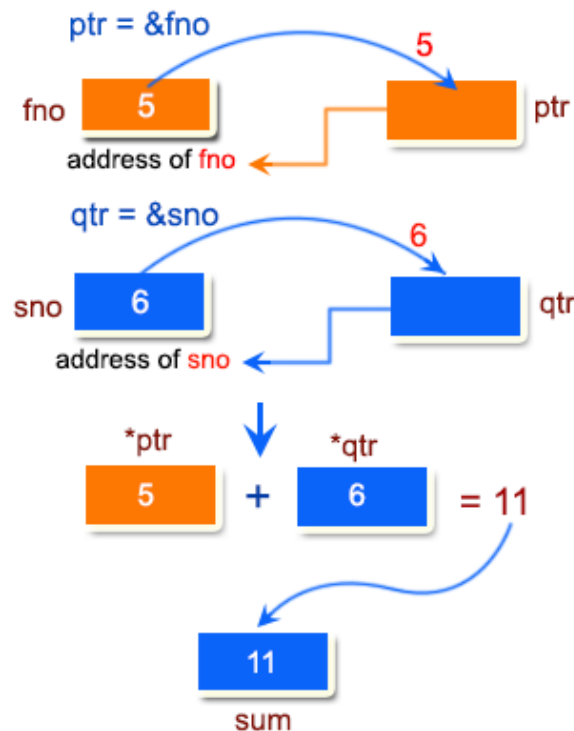
-----  
value at address of m = 300

value at address of fx= 300.600006

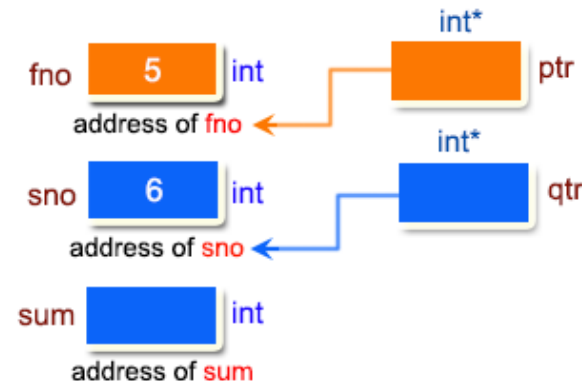
value at address of cht= z

# Bài tập 11.10

- Write a program in C to add two numbers using pointers.



## Variables



# Bài tập 11.10

- Test Data :
- Input the first number : 5
- Input the second number : 6
- Expected Output :
- The sum of the entered numbers is : 11

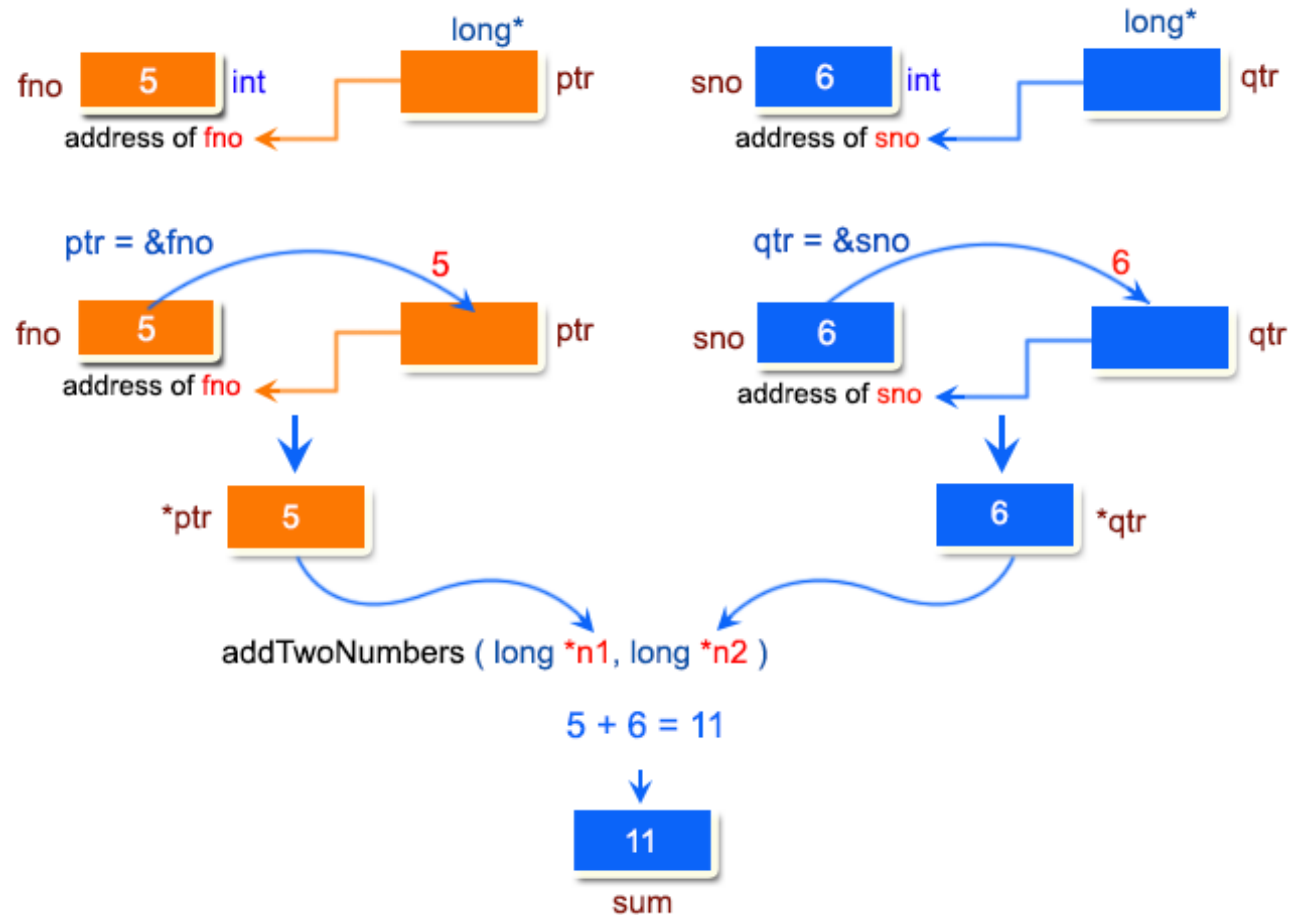
# Bài tập 11.11

- Write a program in C to add numbers using call by reference.
- Test Data :
- Input the first number : 5
- Input the second number : 6
- Expected Output :
- The sum of 5 and 6 is 11



# Bài tập 11.11

- Write a program in C to add numbers using call by reference.

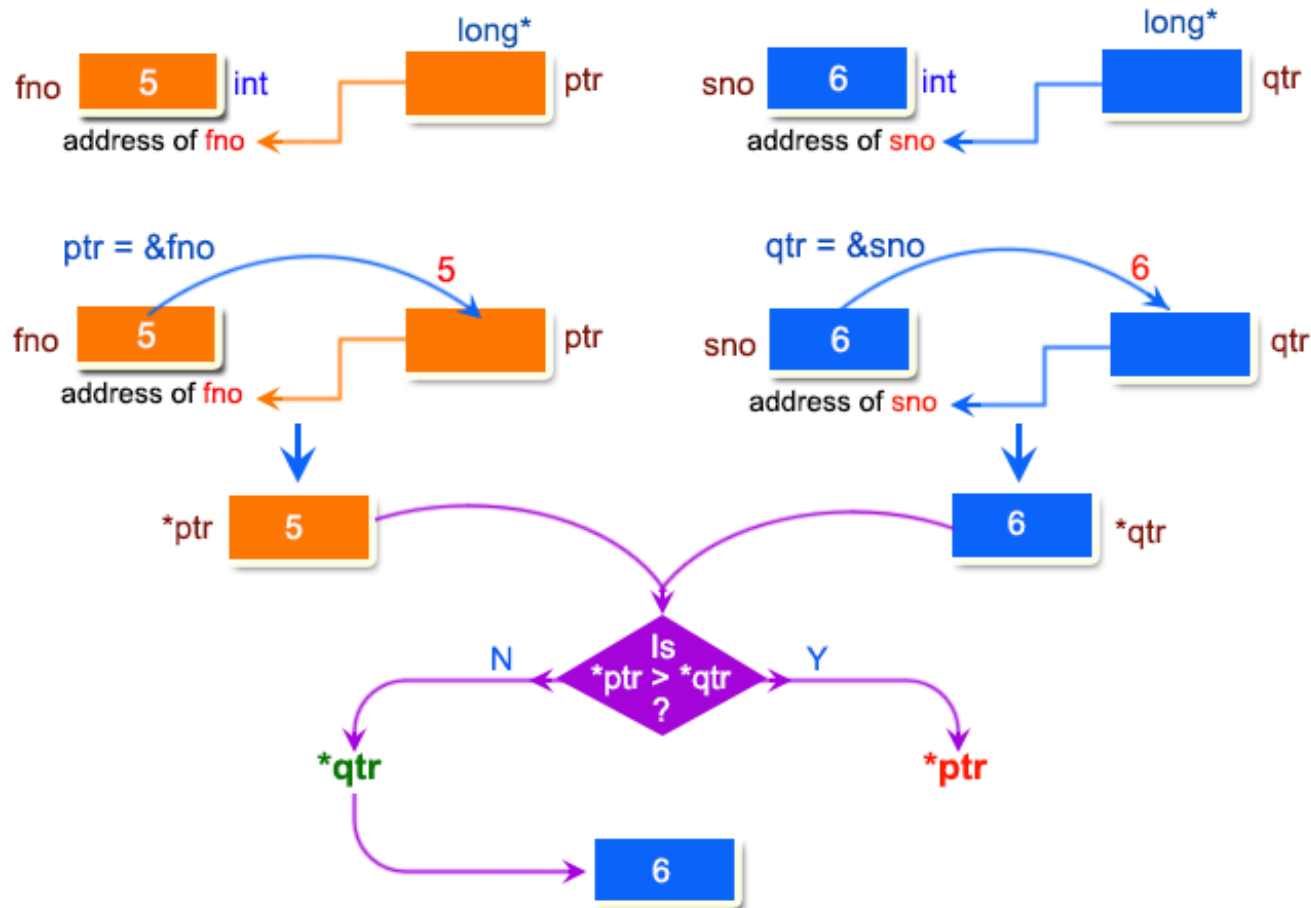


# Bài tập 11.12

- Write a program in C to find the maximum number between two numbers using a pointer.
- Test Data :
- Input the first number : 5
- Input the second number : 6
- Expected Output :
- 6 is the maximum number.

# Bài tập 11.12

- Write a program in C to find the maximum number between two numbers using a pointer.



# Bài tập 11.13

Write a program in C to store n elements in an array and print the elements using pointer. Go to the editor

Test Data :

Input the number of elements to store in the array :5

Input 5 number of elements in the array :

element - 0 : 5

element - 1 : 7

element - 2 : 2

element - 3 : 9

element - 4 : 8

Expected Output :

The elements you entered are :

element - 0 : 5

element - 1 : 7

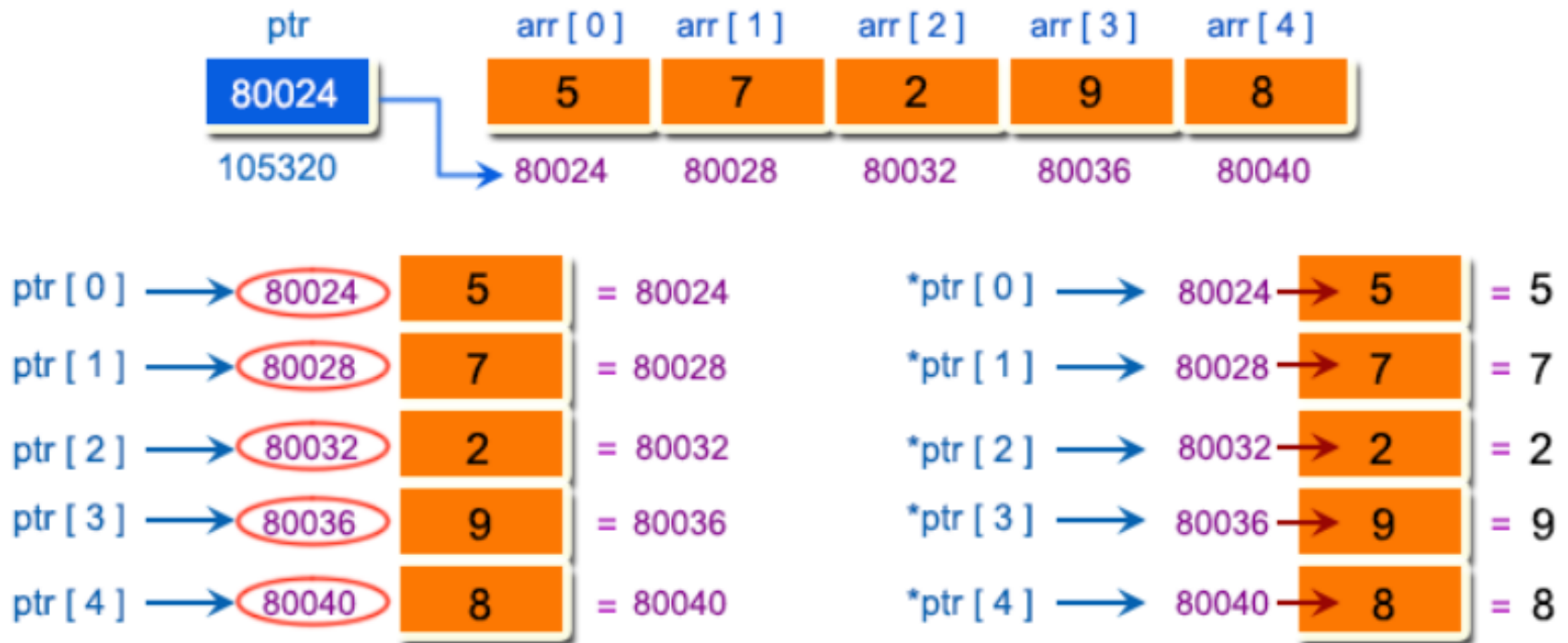
element - 2 : 2

element - 3 : 9

element - 4 : 8

# Bài tập 11.13

- Write a program in C to store n elements in an array and print the elements using pointer



# Bài tập 11.14

- Write a program in C to print all permutations of a given string using pointers. Expected Output :
- The permutations of the string are :
- abcd abdc acbd acdb adcb adbc bacd badc bcad  
bcda bdca bdac cbad cbda cabd cadb cdab cdba  
dbca dbac dcba dcab dacb dabc

# Bài tập 11.15

Write a program in C to find the largest element using Dynamic Memory Allocation. Go to the editor

Test Data :

Input total number of elements(1 to 100): 5

Number 1: 5

Number 2: 7

Number 3: 2

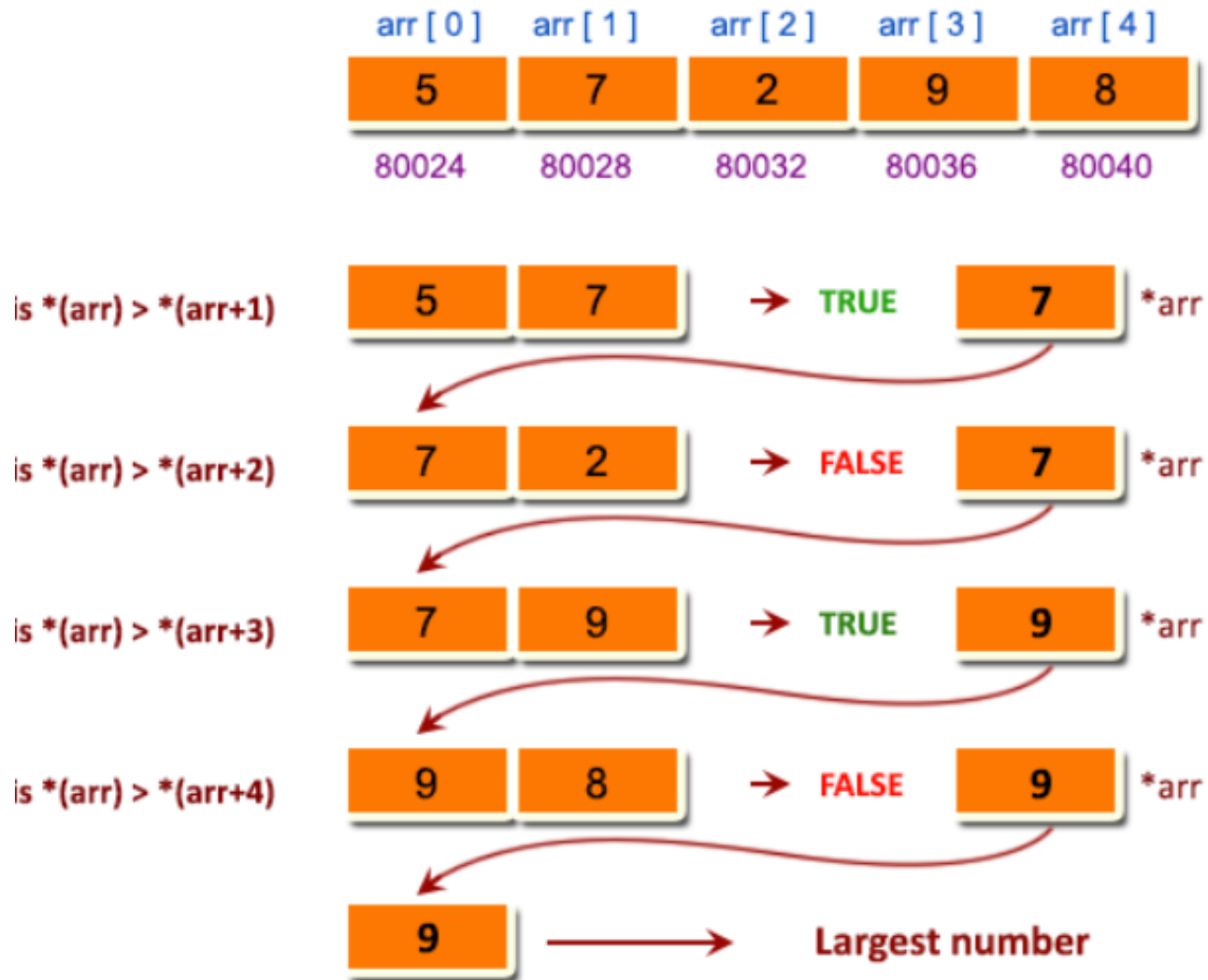
Number 4: 9

Number 5: 8

Expected Output :

The Largest element is : 9.00

# Bài tập 11.15



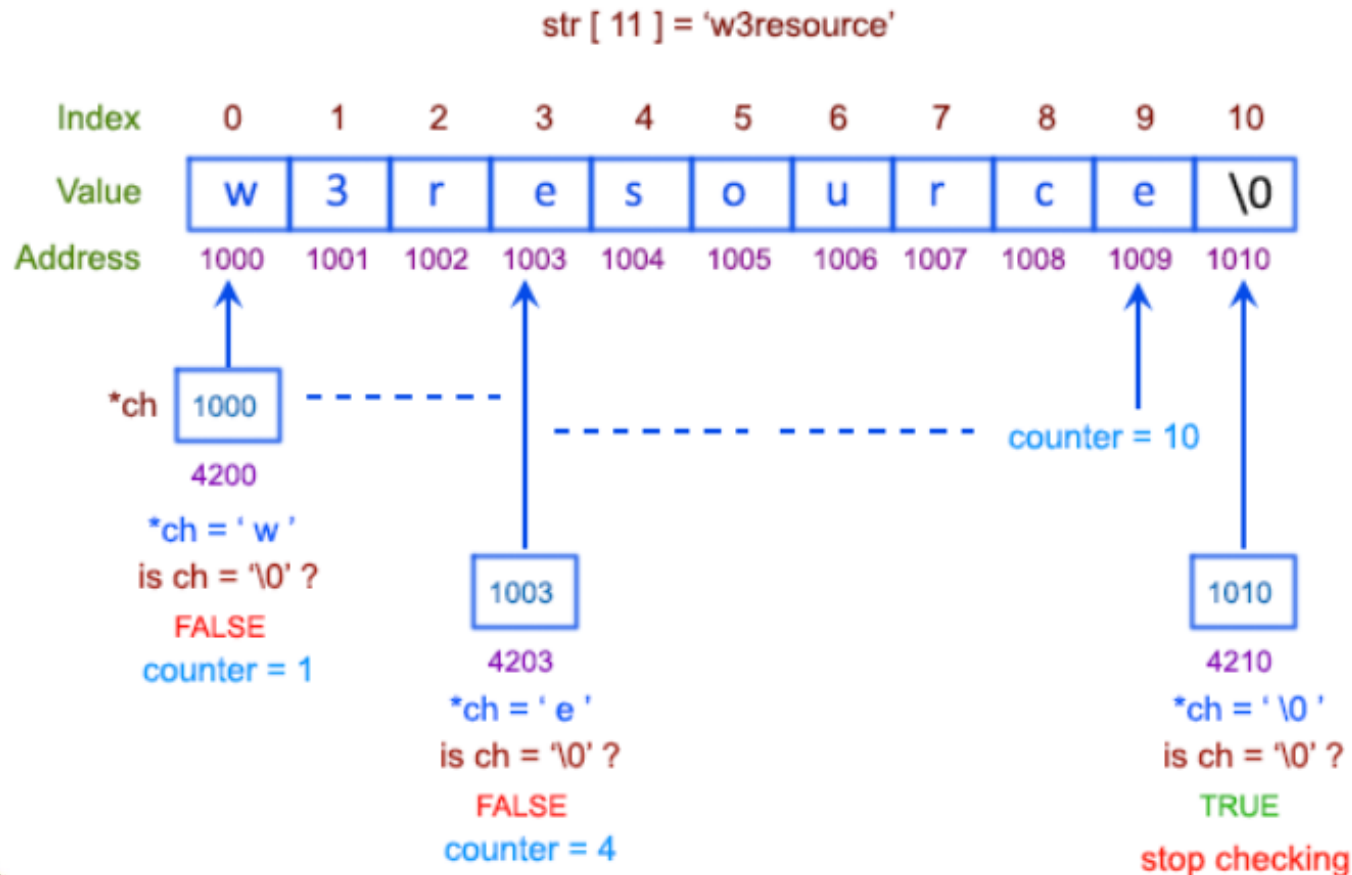


# Bài tập 11.16

- Write a program in C to Calculate the length of the string using a pointer.
- Test Data :
- Input a string : abc
- Expected Output :
- The length of the given string abc is : 3

# Bài tập 11.16

- Write a program in C to Calculate the length of the string using a pointer.



# Bài tập 11.17

- Write a program in C to swap elements using call by reference. Test Data :

- Input the value of 1st element : 5
- Input the value of 2nd element : 6
- Input the value of 3rd element : 7
- Expected Output :
- The value before swapping are :

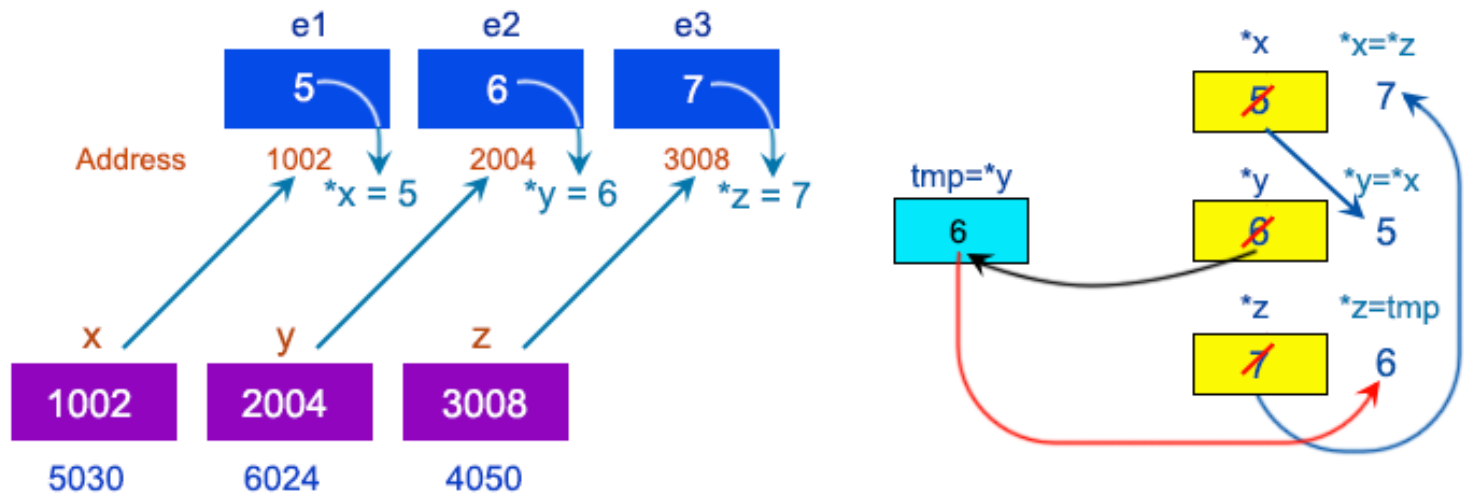
- element 1 = 5
- element 2 = 6
- element 3 = 7

The value after swapping are :

element 1 = 7  
element 2 = 5  
element 3 = 6

# Bài tập 11.17

- Write a program in C to swap elements using call by reference.

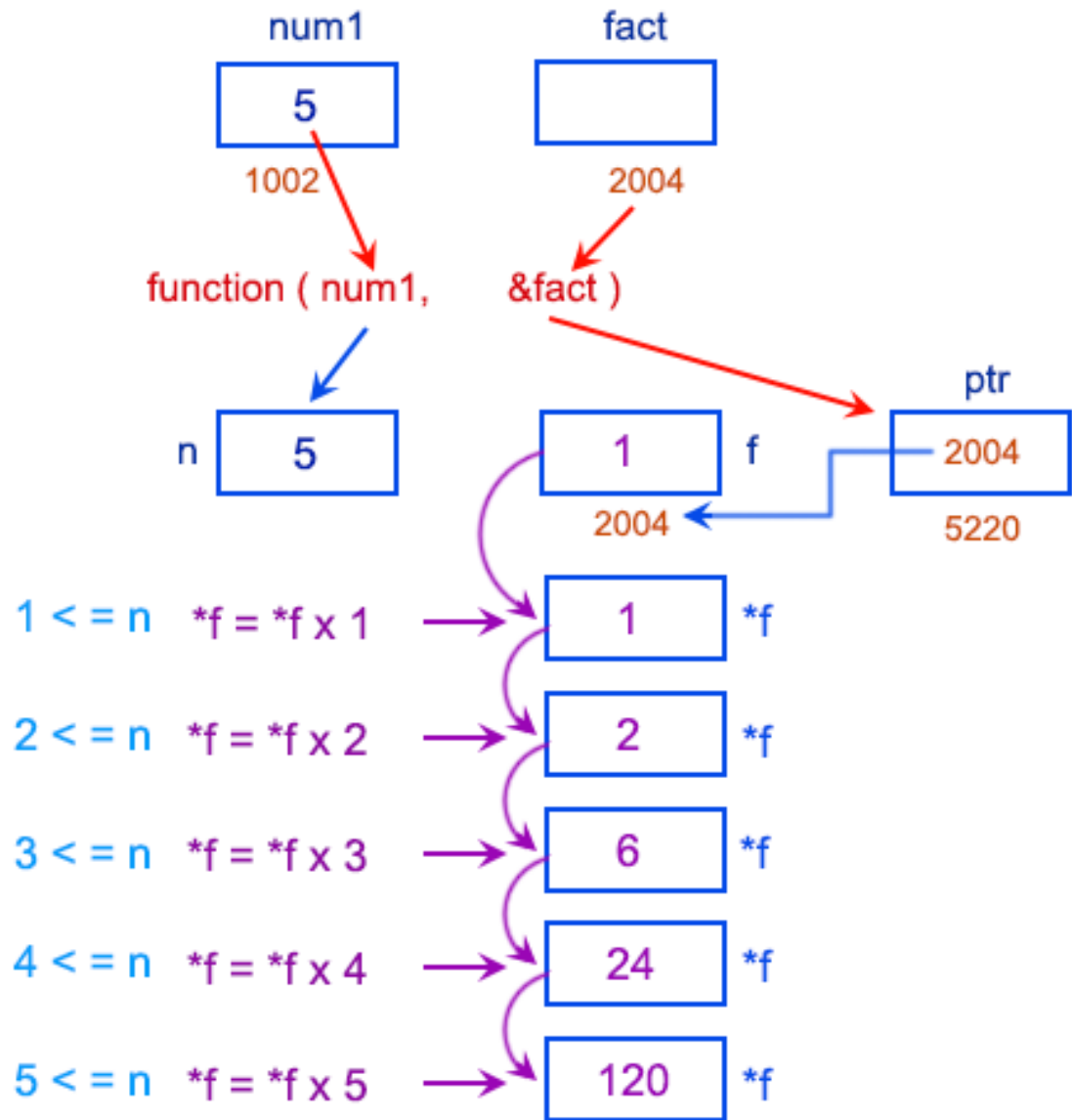


# Bài tập 11.18

- Write a program in C to find the factorial of a given number using pointers.
- Test Data :
- Input a number : 5
- Expected Output :
- The Factorial of 5 is : 120

# Bài tập 11.18

- Write a program in C to find the factorial of a given number using pointers.

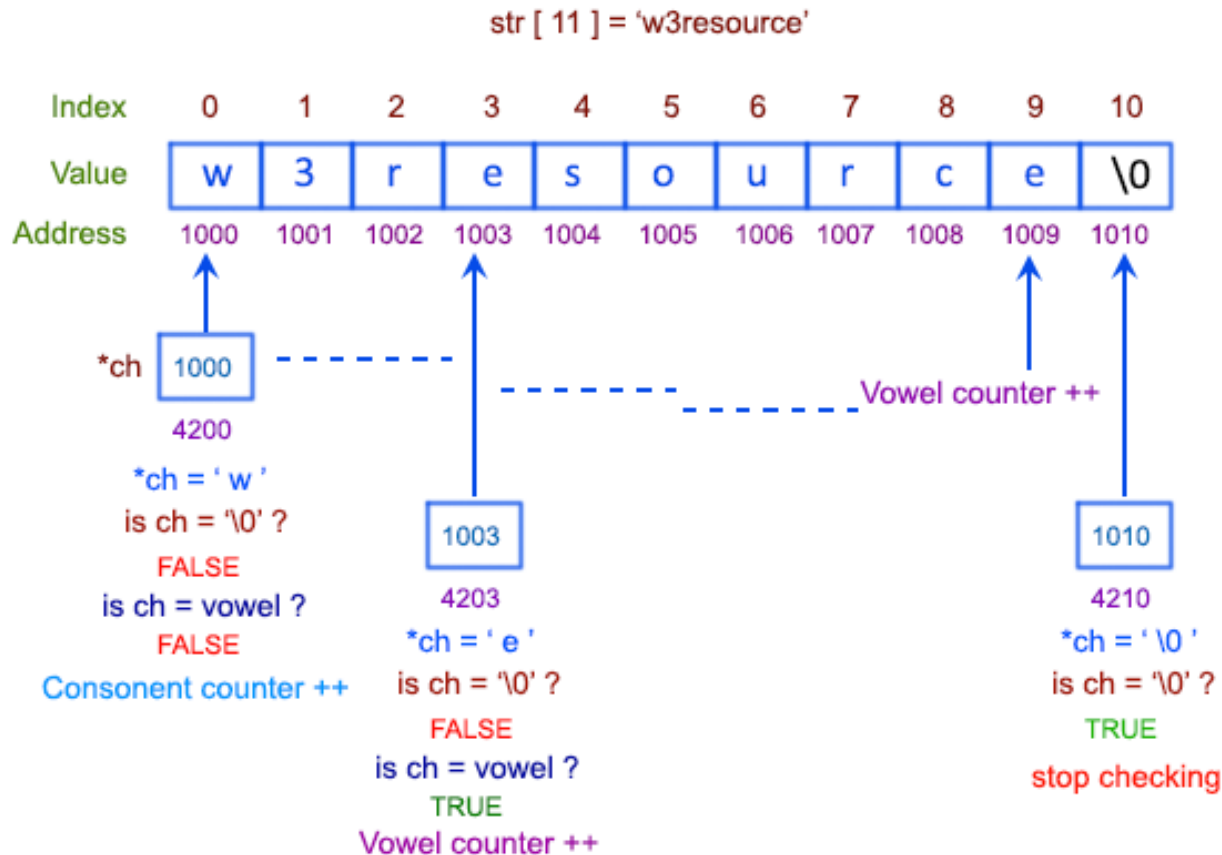


# Bài tập 11.19

- Write a program in C to count the number of vowels and consonants in a string using a pointer.
- Test Data :
- Input a string: string
- Expected Output :
- Number of vowels : 1
- Number of constant : 5

# Bài tập 11.19

- Write a program in C to count the number of vowels and consonants in a string using a pointer.





# Bài tập 11.20

Write a program in C to sort an array using Pointer. Go to the editor

Test Data :

testdata

Expected Output :

Test Data :

Input the number of elements to store in the array : 5

Input 5 number of elements in the array :

element - 1 : 25

element - 2 : 45

element - 3 : 89

element - 4 : 15

element - 5 : 82

Expected Output :

The elements in the array after sorting :

element - 1 : 15

element - 2 : 25

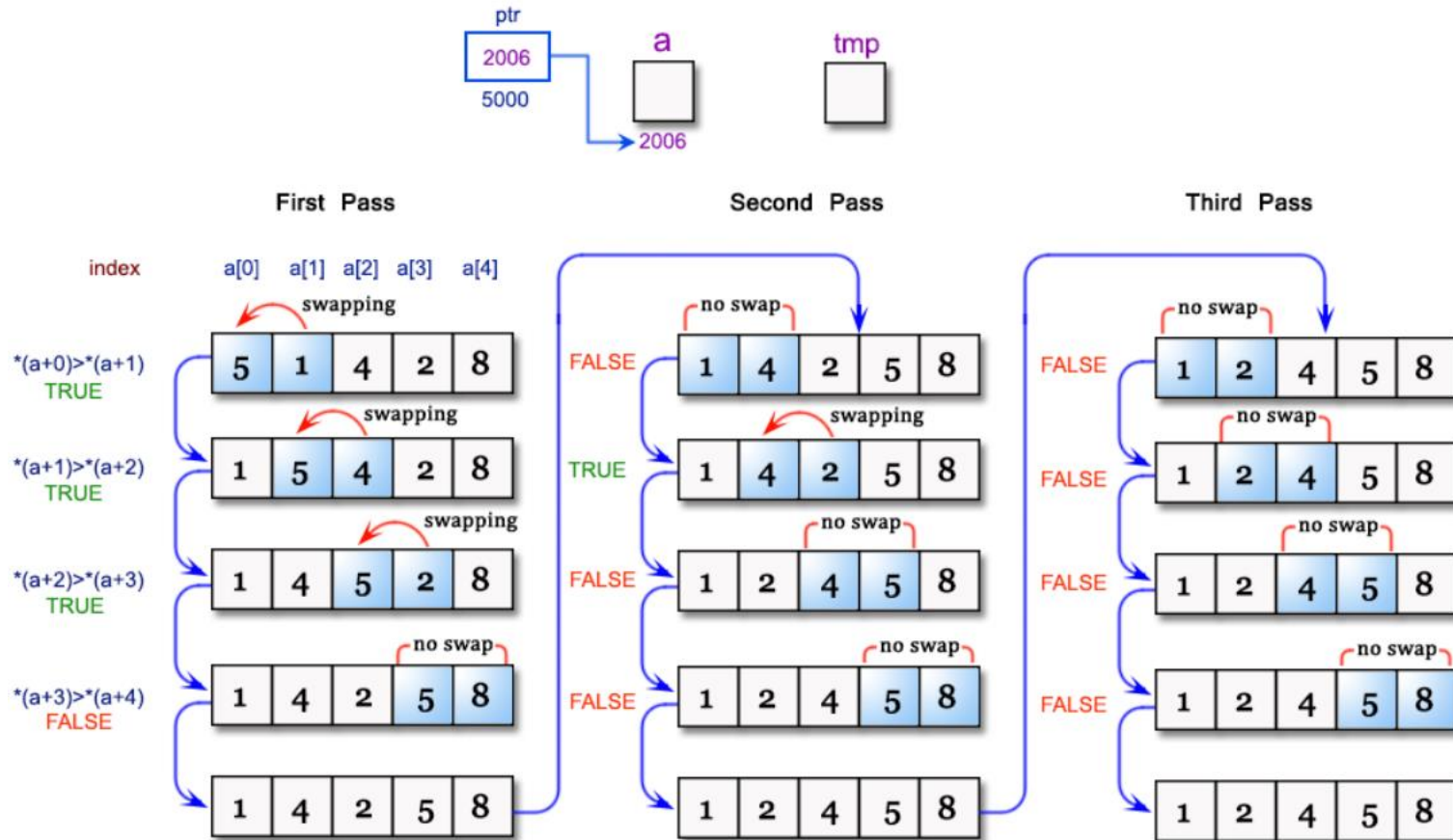
element - 3 : 45

element - 4 : 82

element - 5 : 89

# Bài tập 11.20

- Write a program in C to sort an array using Pointer.

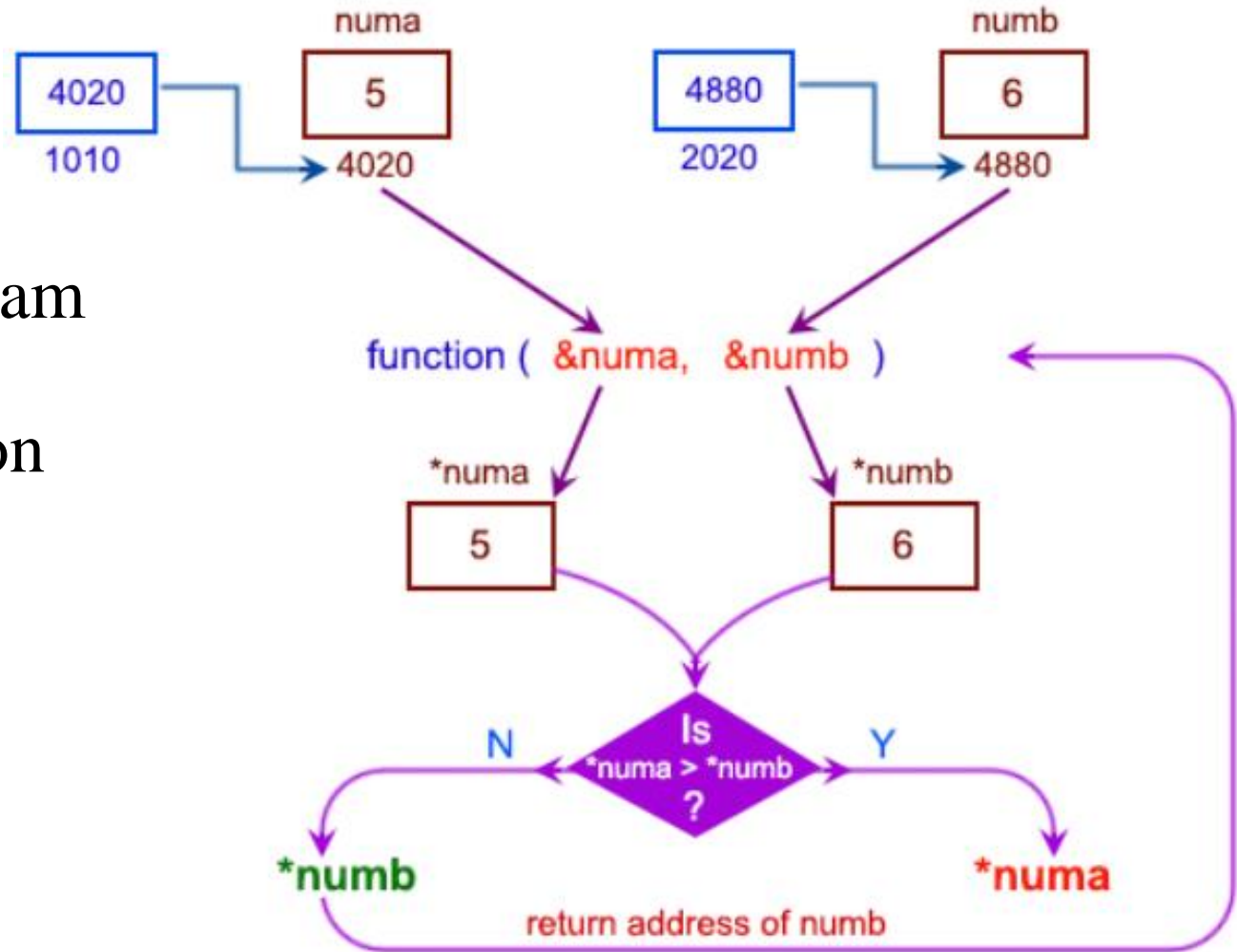


# Bài tập 11.21

- Write a program in C to show how a function returning pointer.
- Test Data :
- Input the first number : 5
- Input the second number : 6
- Expected Output :
- The number 6 is larger.

# Bài tập 11.21

- Write a program in C to show how a function returning pointer.



# Bài tập 11.22

Write a program in C to compute the sum of all elements in an array using pointers.

Test Data :

Input the number of elements to store in the array (max 10) : 5

Input 5 number of elements in the array :

element - 1 : 2

element - 2 : 3

element - 3 : 4

element - 4 : 5

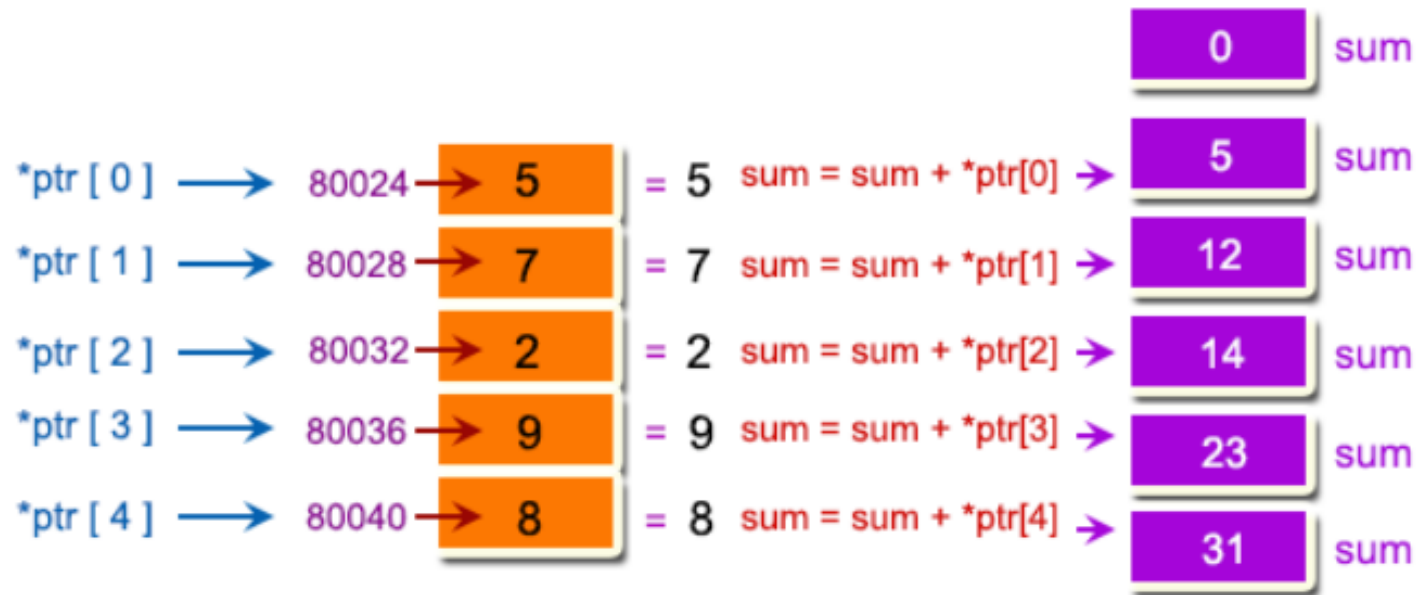
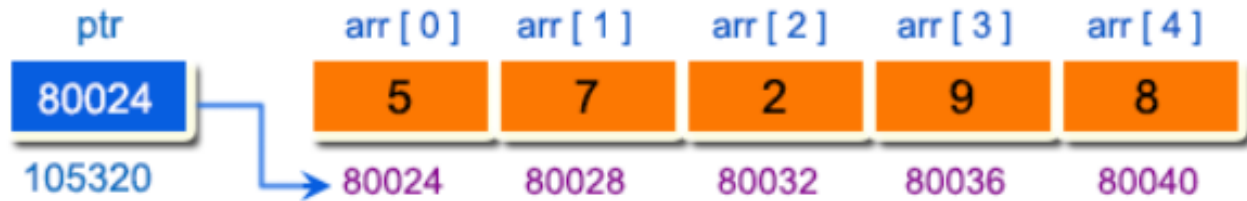
element - 5 : 6

Expected Output :

The sum of array is : 20

# Bài tập 11.22

- Write a program in C to compute the sum of all elements in an array using pointers.



# Bài tập 11.23

Write a program in C to print the elements of an array in reverse order.

Test Data :

Input the number of elements to store in the array (max 15) : 5

Input 5 number of elements in the array :

element - 1 : 2

element - 2 : 3

element - 3 : 4

element - 4 : 5

element - 5 : 6

Expected Output :

The elements of array in reverse order are :

element - 5 : 6

element - 4 : 5

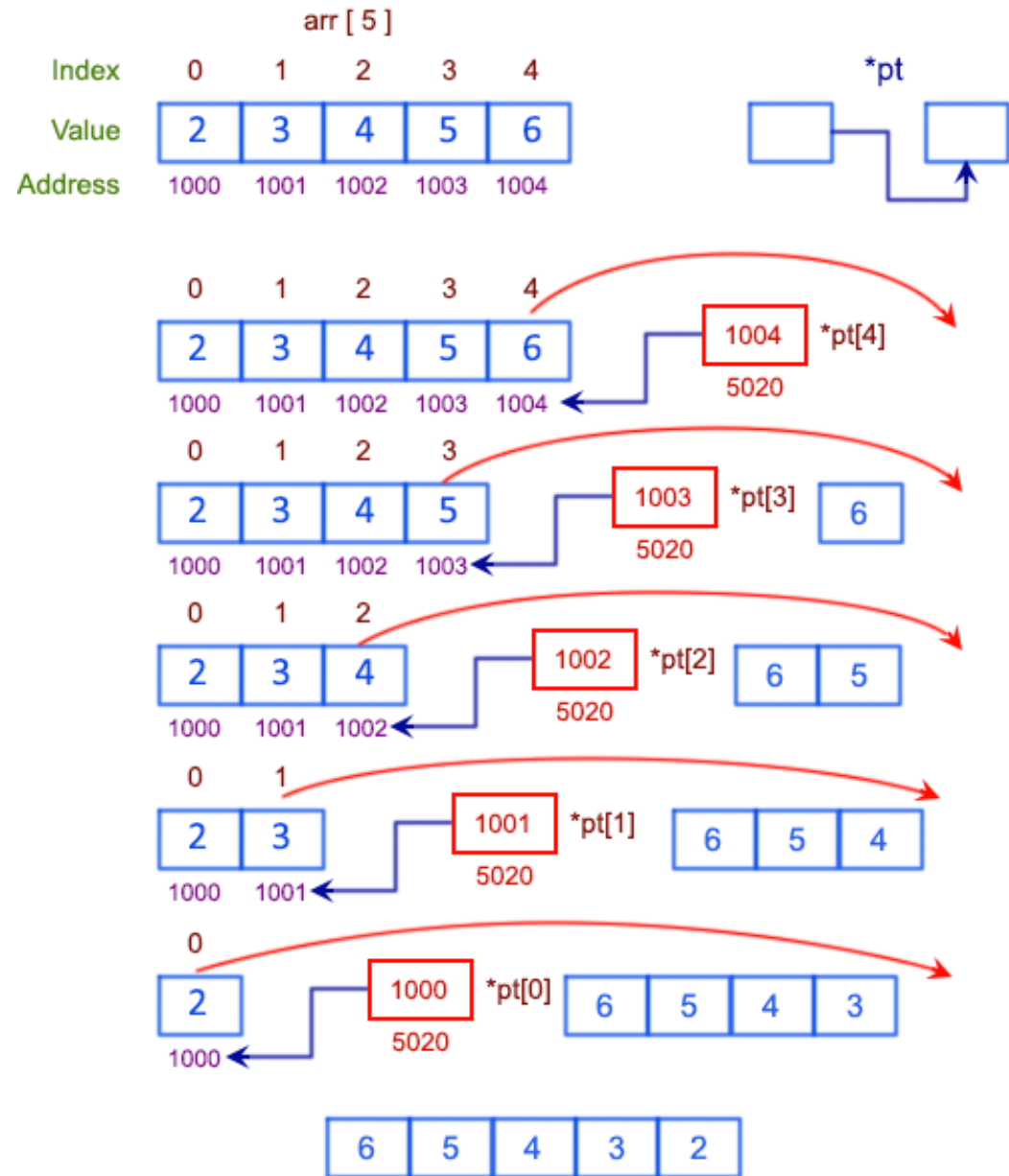
element - 3 : 4

element - 2 : 3

element - 1 : 2

# Bài tập 11.23

- Write a program in C to print the elements of an array in reverse order.



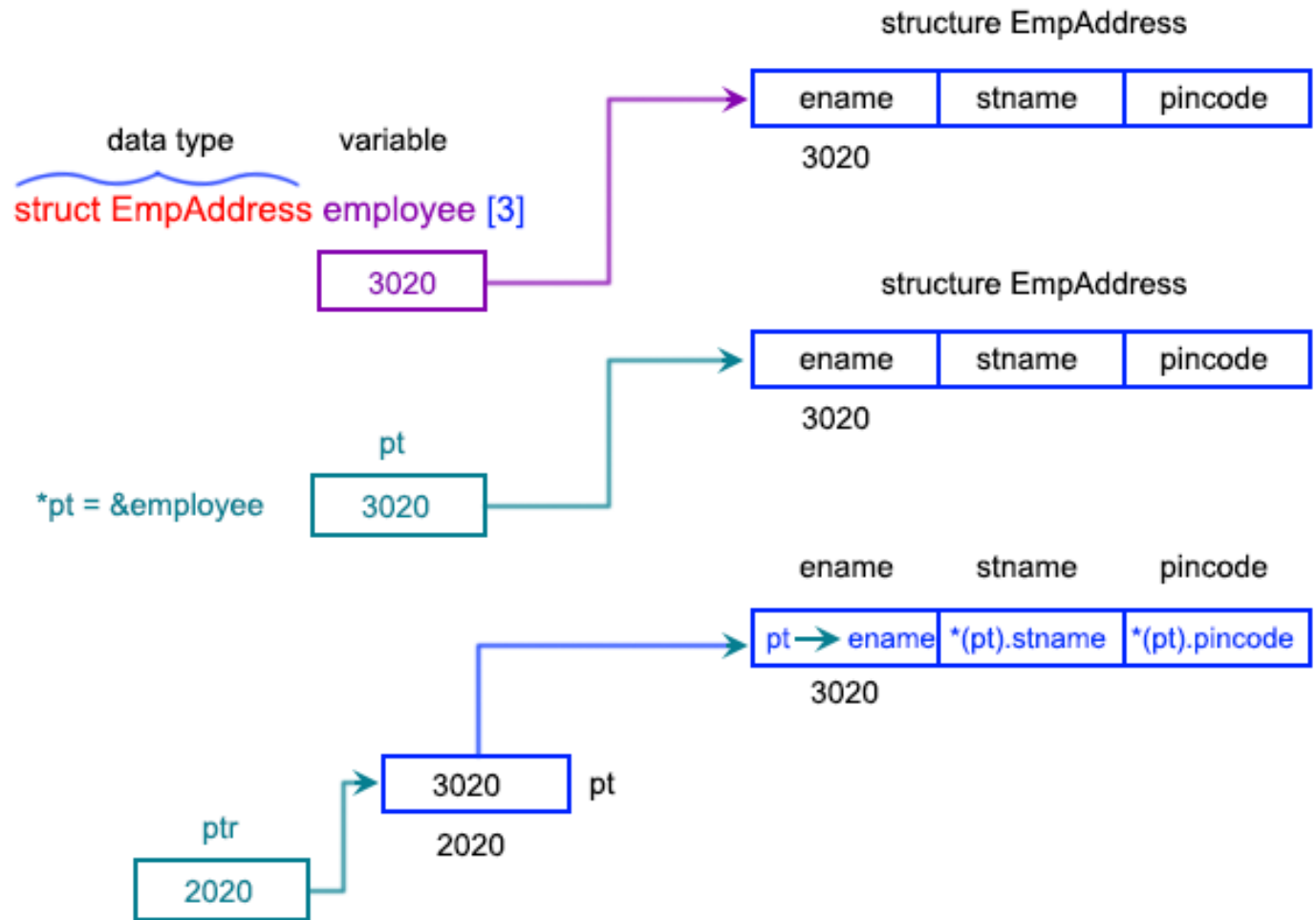


# Bài tập 11.24

- Write a program in C to show the usage of pointer to structure.
- Expected Output :
- John Alter from Court Street

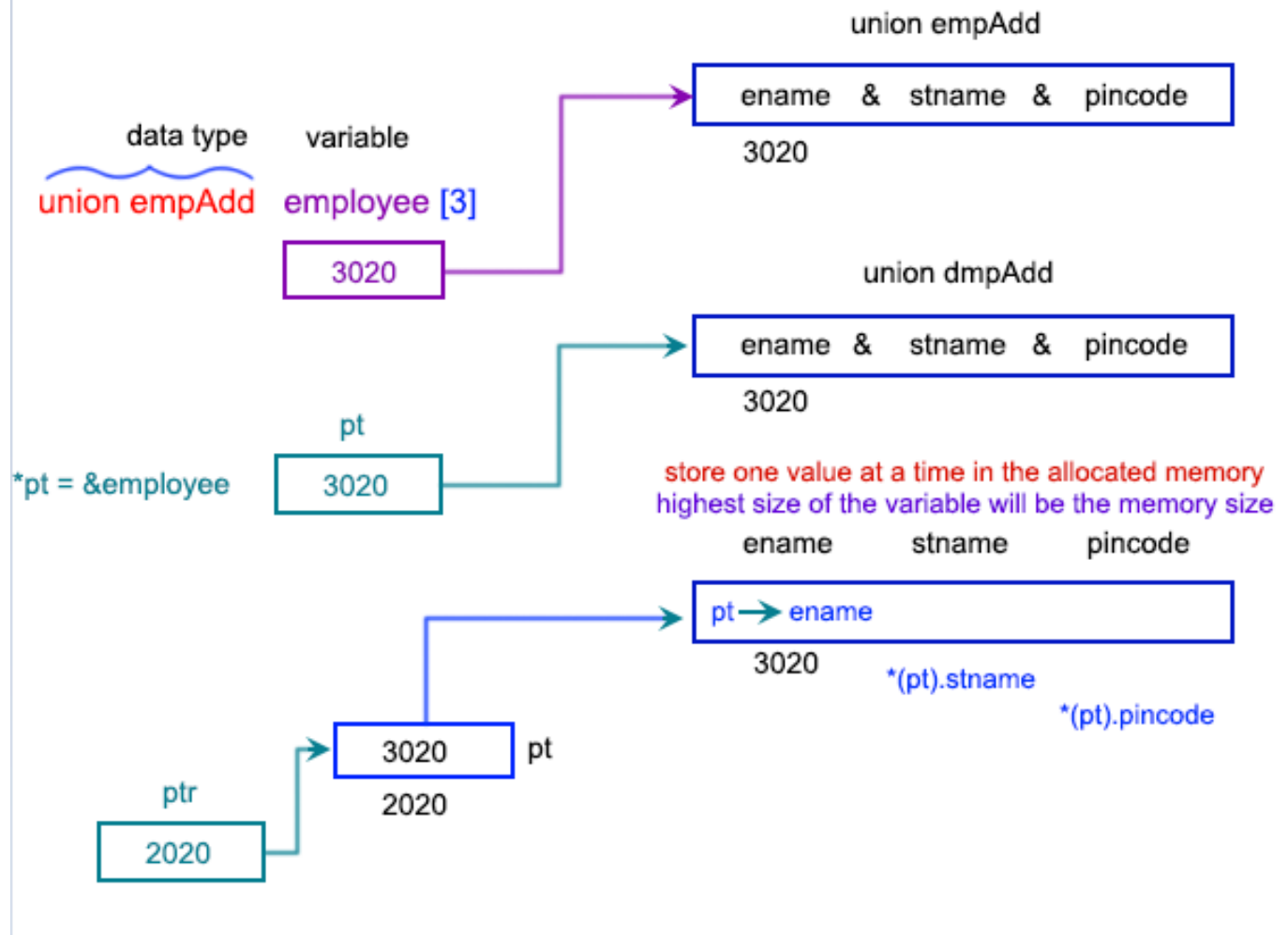
# Bài tập 11.24

- Write a program in C to show the usage of pointer to structure.



# Bài tập 11.25

- Write a program in C to show a pointer to union.
- Expected Output :
- Jhon Mc  
Jhon Mc

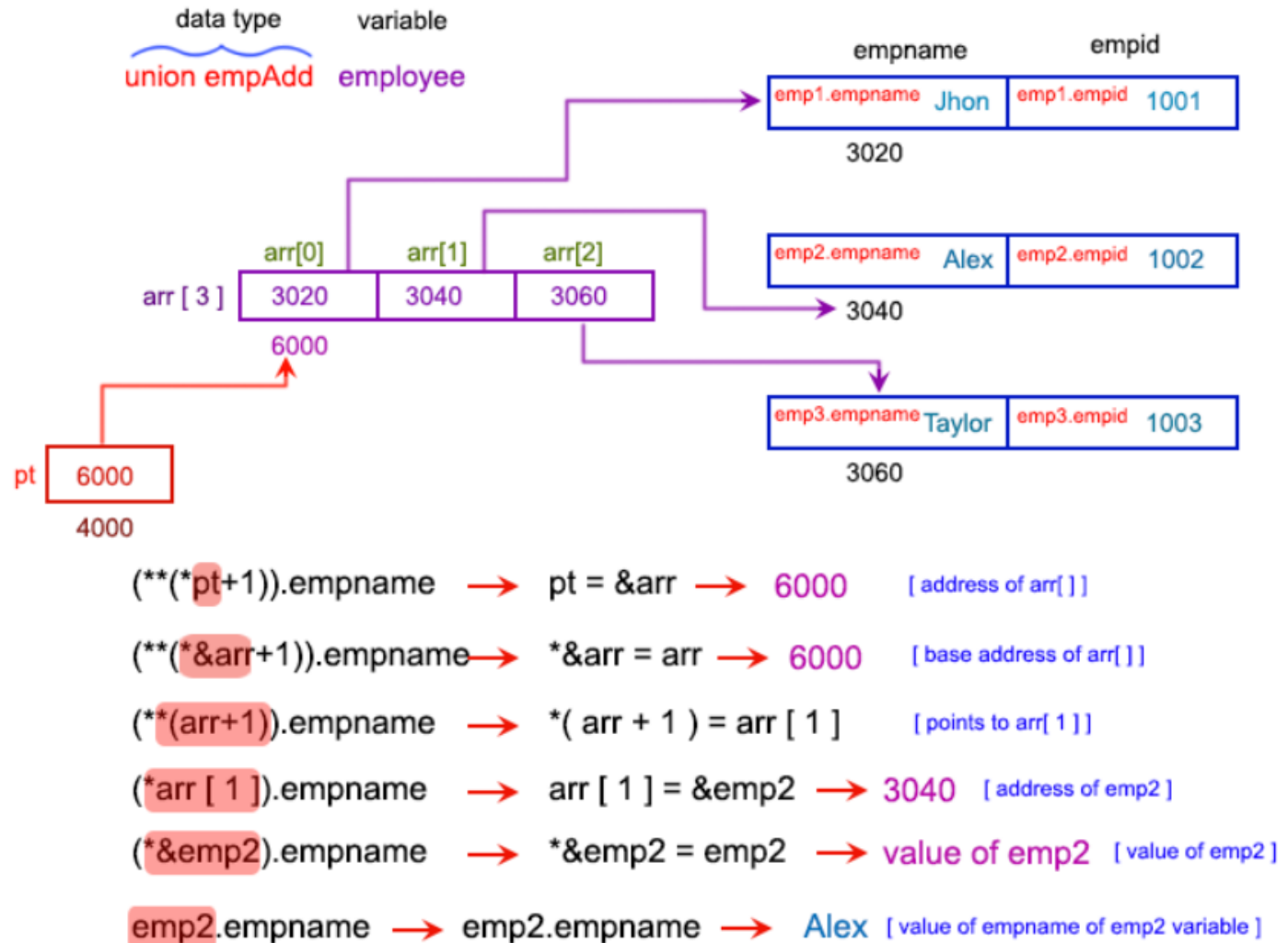


# Bài tập 11.26

- Write a program in C to show a pointer to an array which contents are pointer to structure.
- Expected Output :
- Exmployee Name : Alex
- Employee ID : 1002

# Bài tập 11.26

- Write a program in C to show a pointer to an array which contents are pointer to structure.

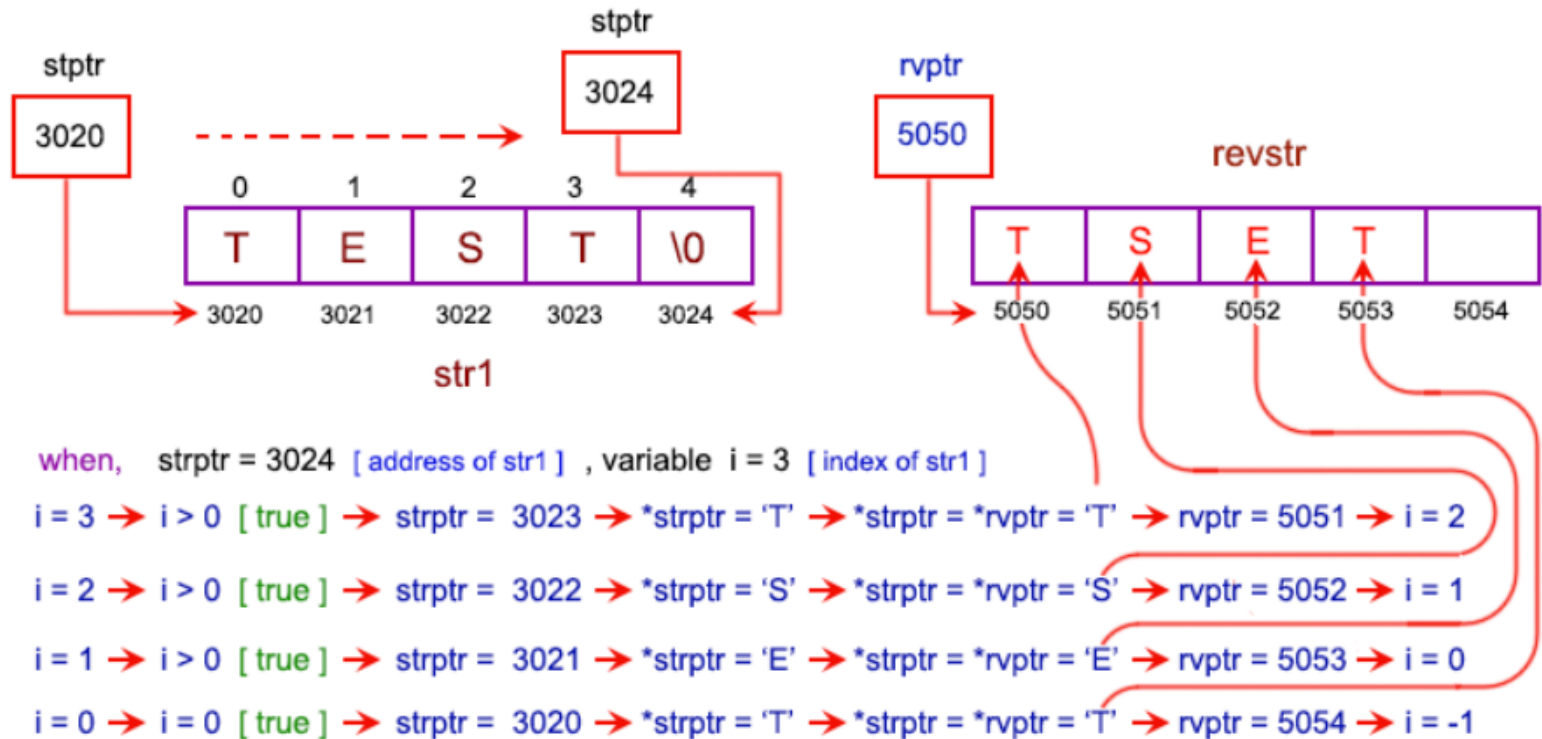


# Bài tập 11.27

- Write a program in C to print all the alphabets using a pointer.
- Expected Output :
- The Alphabets are :
- A B C D E F G H I J K L M N O P Q R  
S T U V W X Y Z

# Bài tập 11.27

- Write a program in C to print all the alphabets using a pointer.



# Bài tập 11.28

- Write a program in C to print a string in reverse using a pointer.
- Test Data :
- Input a string : w3resource
- Expected Output :
- Pointer : Print a string in reverse order :
- -----
- Input a string : abcd
- Reverse of the string is : dcba





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/](http://soict.hust.edu.vn/)



[fb.com/groups/soict](https://fb.com/groups/soict)

