

Programming Fundamental

Basic C Programming

Pointer

ในหน่วยความจำ...

ข้อมูลแต่ละไบต์จะมี ตำแหน่ง (Address)
ในการเก็บข้อมูล

หน่วยความจำ 2 GB

? : มีกี่ Address

? : Address มีขนาดอย่างน้อย กี่บิต

Address Contents

0	01010011
1	01110101
2	01110011
3	01100001
4	01101110
	⋮
n-1	01000011

Address	Data
...	...
N	i
N+1	
N+1	
N+3	
....	...

ตัวแปรแต่ละตัวที่ประกาศ

จะเก็บในหน่วยความจำ

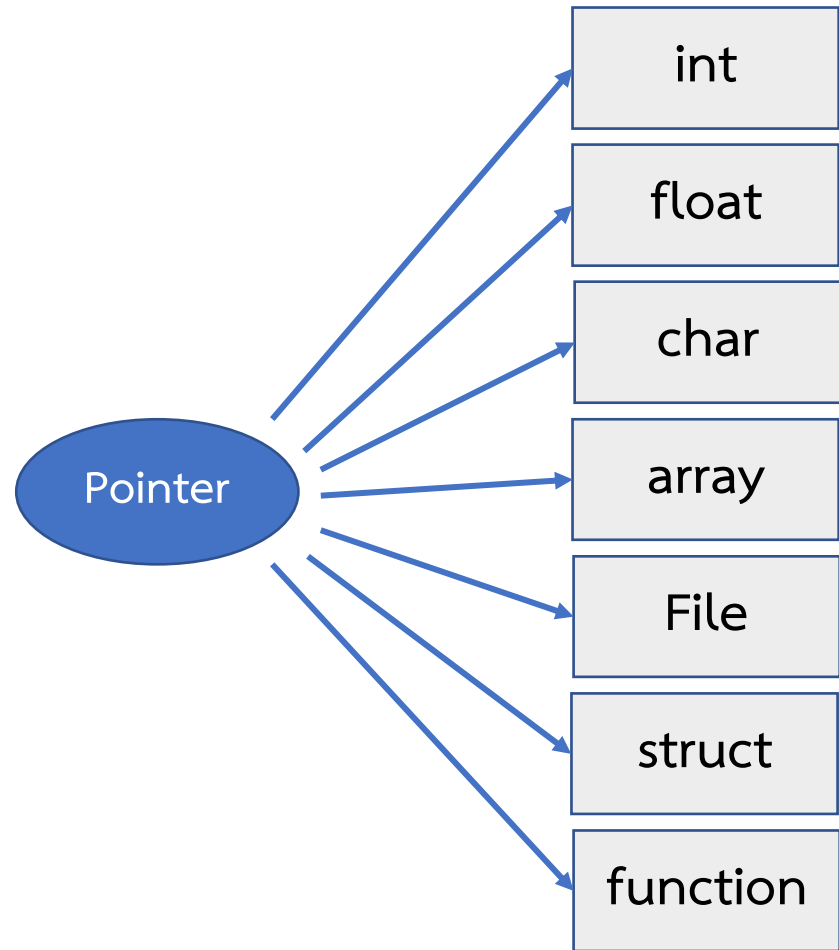
ซึ่งระบบปฏิบัติการจะเลือก Address
ให้โดยอัตโนมัติ

เมื่อการประกาศตัวแปร `int i;`
จะมีการจองพื้นที่ในหน่วยความจำ
จำนวน 4 bytes

จากการเก็บข้อมูลต่างๆ ในระบบ
ที่มี Address ระบุเสมอ
จึงสามารถเข้าถึงข้อมูลต่างๆ
ผ่าน Address ได้

สร้างตัวแปรมาเก็บข้อมูล Address ของ
ข้อมูล ก็สามารถเรียกใช้ข้อมูลนั้นๆ ได้

ตัวแปรเก็บ Address = Pointer

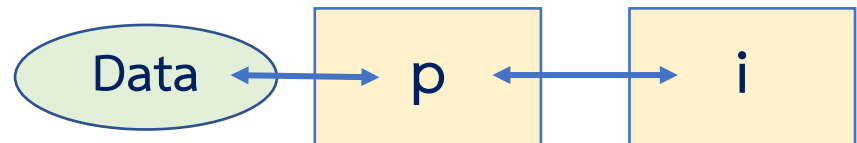
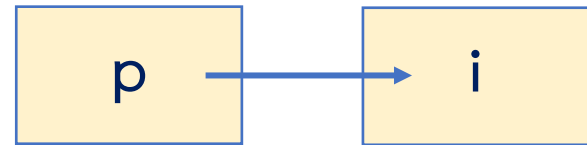


การใช้ Pointer ต้องทำดังนี้

1. สร้างตัวแปร Pointer
2. กำหนดค่าให้ Pointer ไปชี้ที่ตัวแปรอื่นๆ
3. ใช้ข้อมูลที่ Pointer นั้นๆ ชี้อยู่

ตัวแปร Pointer เป็นตัวแปร
ขนาด 4 bytes

*p



1. สร้างตัวแปร Pointer

type *pointer_name

จะได้ตัวแปร pointer_name ที่ชี้ตัวแปรอื่นๆ ที่มีชนิดเป็น type

เครื่องหมาย * หน้าตัวแปร เป็นการกำหนดว่าตัวแปรนั้นเป็น pointer

```
int i=10;
```

```
int *p;
```

p

i

2. กำหนดค่า Pointer ให้ชี้ที่ตัวแปรอื่นๆ

การชี้ในตัวแปร Pointer คือการกำหนดค่า Address ของตัวแปรอื่นๆ
เก็บไว้ในตัวแปร Pointer

อยากได้ Address ของตัวแปรไหน ให้เติม & หน้าตัวแปรนั้น

```
int i=10;  
int *p;  
p=&i;
```

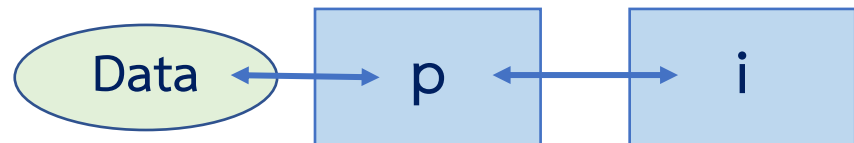


3. การใช้งานข้อมูลที่ Pointer ชี้

เมื่อ p เป็นตัวแปร Pointer
*p คือ ข้อมูลของตัวแปร ที่ p ชี้อยู่

```
int i=10;  
int *p;  
p=&i;  
printf("%d\n",i);  
printf("%d",*p);
```

*p เป็นตัวแทนของ i
ผลลัพธ์การแสดงผล = 10



#Activity

ที่ A , B ตัวแปรต่างๆ สัมพันธ์กันอย่างไร และจะแสดงผลอะไร

```
int x=10,y=20,z=30,*p1,*p2;
```

```
    p1=&x
```

```
    p2=&y
```

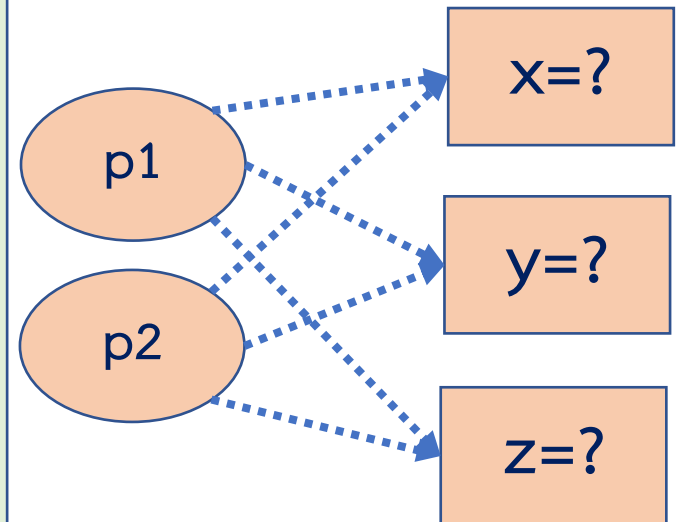
```
printf(“%d %d %d %d %d\n”,x,y,z,*p1,*p2);
```

A

```
    x=80;
```

```
printf(“%d %d %d %d %d\n”,x,y,z,*p1,*p2);
```

B



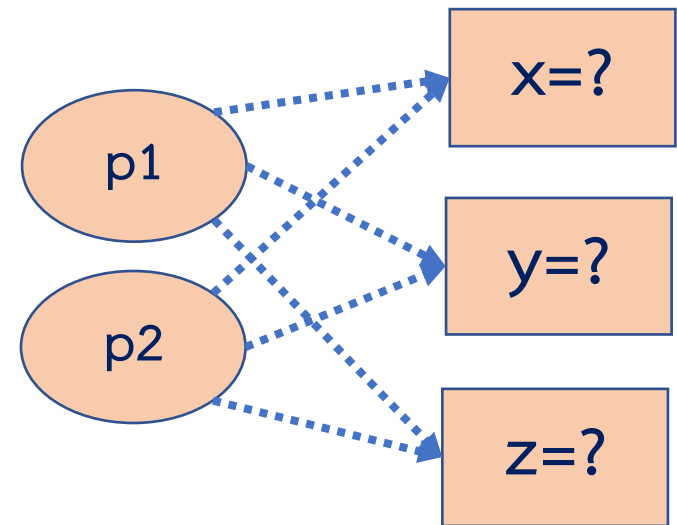
#Activity (ต่อ)

ที่ C , D ตัวแปรต่างๆ สัมพันธ์กันอย่างไร และจะแสดงผลอะไร

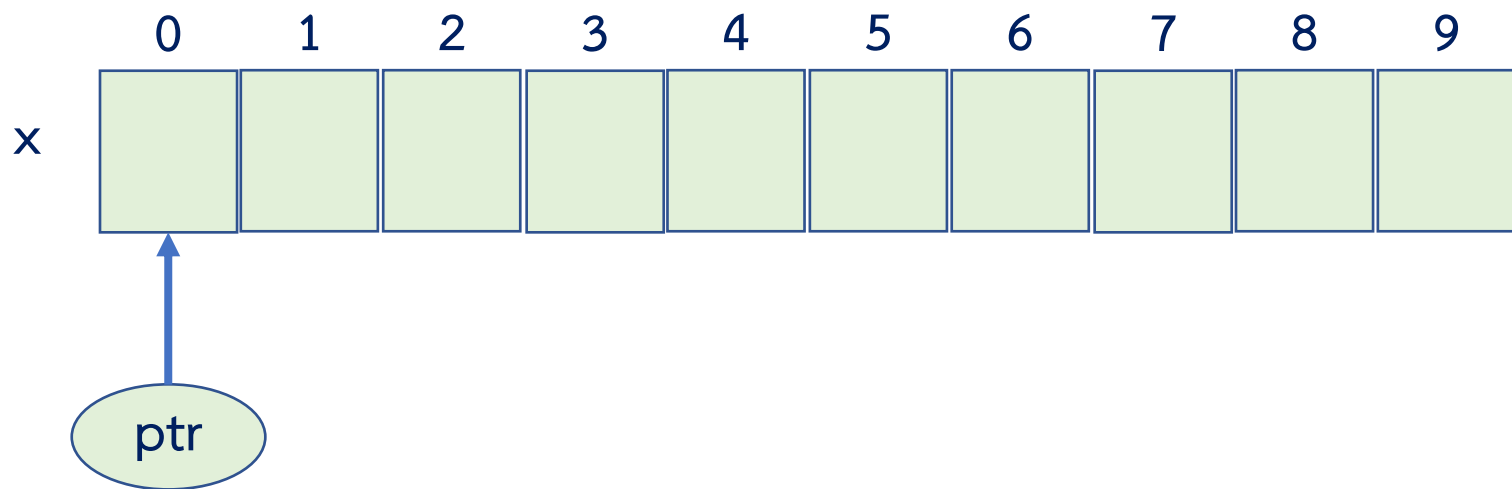
C

`p1=&z``p2=p1;``printf("%d %d %d %d %d\n",x,y,z,*p1,*p2);`

D

`*p1=100;``printf("%d %d %d %d %d\n",x,y,z,*p1,*p2);`

จากการทำงานของ Pointer ทำให้ Pointer สามารถเป็นตัวแทนของตัวแปรอื่นๆ ได้
เช่น การใช้ pointer ชี้ array



ซึ่งการเป็นตัวแทนของตัวแปรอื่นๆ ได้ ต้องสามารถเก็บข้อมูล และนำข้อมูลในตำแหน่ง
ต่างๆ ออกมาใช้ได้เหมือนกับตัวแปร array

การใช้ Pointer กับ Array

1. สร้างตัวแปร Pointer สำหรับชี้ Array กำหนดชนิดให้ตรงกับ Array

```
int i[5]={30,40,50,60,70};
```

```
int *p;
```

2. กำหนดค่าให้ Pointer ไปชี้ที่ตัวแปร array

```
p=i;
```

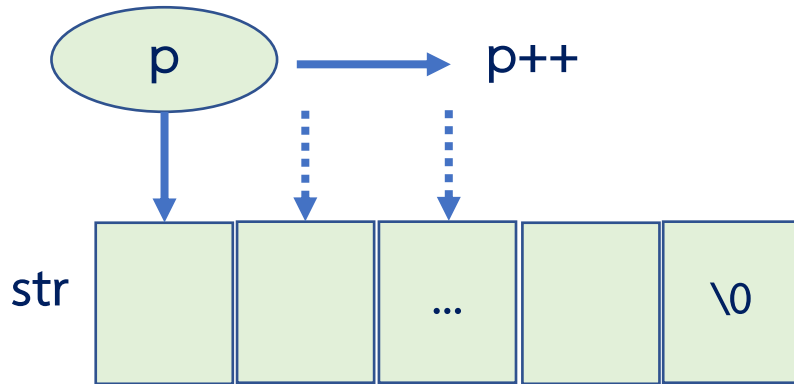
3. การรับค่าและแสดงผลใช้ *(p+index) สำหรับอ้างอิง i[index]

```
printf(“%d %d\n”,i[2],*(p+2));
```

ไม่มี & นำหน้า i

เพราะตัวแปร array ถือว่าเป็น
static pointer

ตัวอย่างการใช้ Pointer กับ String



โปรแกรมจะรับ String 1 ตัว
ยาวไม่เกิน 50 ตัวอักษร
แล้วแสดงทีละตัวอักษรต่อ 1 บรรทัด

สังเกตการใช้ pointer กับ string
และ loop ที่อ่าน string

```
#define _CRT_SECURE_NO_WARNINGS 1
#include<stdio.h>
#include<iostream>
int main()
{
    char *p, str[50];

    p = str;
    scanf("%s", str);

    while (*p != '\0') {
        printf("%c\n", *p);
        p++;
    }

    system("pause");
}
```

#Activity

ให้นักศึกษาเขียนโปรแกรมรับ input เป็น string
แล้วแสดงผล string ที่รับมา โดยใช้ pointer

- เฉพาะตัวอักษร a-z,A-Z
- แสดงผลโดยเปลี่ยนตัวพิมพ์เล็กเป็นตัวพิมพ์ใหญ่
ตัวพิมพ์ใหญ่เป็นตัวพิมพ์เล็ก

Input : aB\$Cde%f#

Output : AbcDEF

#Activity

Palindrome เป็นคุณสมบัติของ string ที่สมมาตรกันซ้ายขวาเช่น
“aba” , “atoyota” , “1234321” , ...
ให้นักศึกษาเขียนโปรแกรมตรวจสอบว่า string ที่รับจากผู้ใช้งาน
เป็น palindrome หรือไม่

Input : abcdcba
Output : Palindrome

Input : teststring
Output : Not Palindrome

#Activity

uniq() เป็นการตัดตัวอักษรที่อยู่ติดกันและซ้ำกันให้เหลือเพียงตัวเดียว

`uniq("aaabbbbbc")="abc"`

`uniq("xxaaxabcyyypp")="xaxabcyp"`

ให้นักศึกษาเขียนโปรแกรมรับค่า string 2 ตัว (A,B)

แล้วให้ตรวจสอบว่า `uniq(A)=uniq(B)` หรือไม่

โดยใช้ pointer กับ array

Hint : ตัวอักษรตัวสุดท้ายของ string ใดๆ ในภาษา C คือ `'\0'`

เทียบ *Aptr กับ *Bptr

