



C Review for 305214 / 235012

Fundamental of Data Structures
and Algorithms

• • • | Topic

- Data type
- Array and String
- Struct and Typedef
- Pointer
- Pointer and Array
- Pointer and Structure

2

• • • | Data Type

- Define Data type

```
DataType Variable_name;
```

```
DataType Variable_name=value;
```

```
DataType Variable_name=value,  
Variable_name=value ;
```

3

• • • | Data Type (essential)

Type		Bytes
char	character variable	1
int	integer variable	2
unsigned	type modifier / unsigned int	2
long	type modifier / long int	4
float	single precision real number	4
double	double precision real variable	8

4

● ● ● | Array and String

○ Array in C

`DataType Variable_name[array_size];`

`Variable_name[0]` to `Variable_name[array_size - 1]`

`Variable_name` → Pointer that point to first address of array.

5

● ● ● | Array and String

○ String in C

String is a single dimension of char array

`char String_name[string_size];`

****** String must end with "NULL" or '/0' character

6

● ● ● | Struct and Typedef

○ Struct

```
struct name
{
    DataType sub_variable1;
    DataType sub_variable2;
    ...
};
```

```
struct name variable_name;
```

7

● ● ● | Struct and Typedef

○ Struct

```
struct name
{
    DataType sub_variable1;
    DataType sub_variable2;
    ...
}variable_name;
```

8

● ● ● | Struct and Typedef

- Typedef : `typedef DataType type_name;`

```
typedef struct name
{
    DataType sub_variable1;
    DataType sub_variable2;
    ...
} struct_type_name;
```

```
struct_type_name variable_name;
```

9

● ● ● | Stdio.h printf & scanf

- I/O of stdio.h

```
printf( "format", variables);
```

```
scanf( "format", variable_pointers);
```

10

● ● ● | Format Identifiers

- Format for printf() and scanf()

%d	decimal integers
%c	character
%f	float and double number
%s	string
%p	pointer
%x	hex integer

11

● ● ● | ทบทวน

- เขียน โปรแกรมเพื่อรับข้อมูล ชนิดข้อมูลแบบ integer float และ string แล้วให้แสดงผลให้ได้ผลดังนี้

name : **Apiwat**

year : **2**

GPA : **2.0**

My name is **Apiwat**, year **2** student with GPA **2.0**.

12

● ● ● | Pointer

- Pointer keeps (point to) variable address

`DataType *pointer_name;`

- Address of variable

`&variable_name;`

- Value of pointer

`*pointer_name;`

13

● ● ● | Pointer Example

```
void main()
{
    int *a, b;
    b = 2;
    a = &b;
    *a=4;    → b=4
}
```

14

● ● ● | Pointer Example

```
void main()
{
    int *a;
    *a=4;
}
```

Error !!!

15

● ● ● | Function that pass by pointer

- Example

```
void swop(int *a, int *b)
{
    int temp;
    temp = *a;
    *a = *b;
    *b = temp;
}
```

16

●●● | ทบทวน

- เขียน function โดยทำการส่งค่าผ่าน address โดยกำหนดให้มีชื่อและการทำงานดังต่อไปนี้

ชื่อฟังก์ชัน `statv(int *v1, int *v2, int *v3)`

โดยที่จะมีการทำงานโดยเมื่อเรียกทำงานแล้ว ค่าตัวแปรจะเป็นดังต่อไปนี้

`v1` = ผลรวมของตัวแปรทั้ง 3

`v2` = ค่า minimum

`v3` = ค่า maximum

พร้อมทั้งทำการป้อนค่าและแสดงผล

17

●●● | Pointer and Array

- Array name → address of first array variable

```
int n[4], *pn;
{
    pn = n; <--> ( pn=&n[0]; )
    *pn = 1;
    pn++;
    *pn = 2; <--> ( n[0] = 1, n[1] = 2 )
};
```

18

●●● | Pointer and Array

- Example string copy function

```
void string_copy(char *s, char * d)
{
    while(*s!=NULL)
    {
        *d=*s;
        d++; s++;
    }
    d++; *d=NULL;
}
```

19

●●● | Pointer and Structure

- Structure

```
struct coordinate
{
    float x;
    float y;
    float z;
};
```

20

● ● ● | Pointer and Structure

○ Type Define

```
struct coordinate
```

```
{  
    float x;  
    float y;  
    float z;  
};
```

```
typedef struct coordinate COORD;
```

```
typedef struct
```

```
{  
    float x;  
    float y;  
    float z;  
} COORD;
```

21

● ● ● | Pointer and Structure

○ Structure Reference

```
typedef struct
```

```
{  
    float x;  
    float y;  
    float z;  
} COORD;
```

```
COORD point_A;
```

```
point_A.x = 0;
```

```
point_A.y = 0;
```

```
point_A.z = 0;
```

22

● ● ● | Pointer and Structure

○ Pointer to Structure

```
typedef struct
```

```
{  
    float x;  
    float y;  
    float z;  
} COORD;  
COORD A;
```

```
COORD *point_ptr;
```

```
point_ptr = &A;
```

```
(*point_ptr).x=0;
```

```
(*point_ptr).y=0;
```

```
(*point_ptr).z=0;
```

23

● ● ● | Pointer and Structure

○ Pointer to Structure

```
typedef struct
```

```
{  
    float x;  
    float y;  
    float z;  
} COORD;  
COORD A;
```

```
COORD *point_ptr;
```

```
point_ptr = &A;
```

```
point_ptr->x=0;
```

```
point_ptr->y=0;
```

```
point_ptr->z=0;
```

24