

Topic 2: Pointer & structure

Pointer & structure

- Structure declare

```
struct reg  
{  
    int ID;  
    int score;  
}; // must have a semicolon
```

```

#include <stdio.h>
struct person
{
    int age;
    float weight;
};
int main()
{
    struct person *person_ptr, person1; //person1 was declared as a static variable
    person_ptr = &person1;

    printf("Enter age: ");
    scanf("%d", &person1.age );    // to access static member use .
    // scanf("%d", &(person_ptr->age));

    printf("Enter weight: ");
    scanf("%f", &(*person_ptr).weight );
        // can be &((*person_ptr).weight)
        // *person_ptr is the content, (*person_ptr).weight → the weight member

    printf("age: %d, age: %d, weight: %.2f\n", person1.age,
                                                (*person_ptr).age,
                                                person_ptr->weight);
}

```

Pointer & structure

- I like this kind of declaration more. (remember it!)

```
typedef struct reg
{
    int ID;
    int score;
} tReg;
```

```
tReg *stu_ptr;
stu_ptr = (tReg *) malloc (sizeof(tReg));
stu_ptr ->ID = 10;
```

Pointer & structure (quiz)

```
#include <stdio.h>
#include <stdlib.h>
#define N 4

typedef struct reg
{
    int id;
    int score;
}tReg;

int main (void)
{
    tReg *stu_ptr;
    tReg *head;
    int i;
```

```
    stu_ptr = (tReg *) malloc (sizeof(tReg)*N);
    head = stu_ptr;

    stu_ptr->id = 1; stu_ptr->score = 99;
    stu_ptr++;
    stu_ptr->id = 2; stu_ptr->score = 80;

    stu_ptr = head;
    stu_ptr[2].id = 40; stu_ptr[2].score = 60;

    stu_ptr = head;
    for (i = 0; i < N ; i++)
    {
        printf("id: %d with score: %d \n",
                stu_ptr->id, stu_ptr->score);
        stu_ptr++;
    }
    return 0;
}
```

Pointer & structure

```
#include <stdio.h>
#include <stdlib.h>
#define N 4

typedef struct reg
{
    int id;
    int score;
}tReg;

int main (void)
{
    tReg student[N]; //not fashion
    tReg *stu_ptr;
    tReg *head;
    int i;

    student[3].id = 0;
    student[3].score = 99;

    // Assess as a static using •
```

```
stu_ptr = (tReg *) malloc (sizeof(tReg)*N);
head = stu_ptr; //Avoid lost your way. Store the head
```

```
stu_ptr->id = 1;    stu_ptr->score = 99;
stu_ptr++;        // to the next element
stu_ptr->id = 2;    stu_ptr->score = 80;
```

```
stu_ptr = head; //back to the original
stu_ptr[2].id = 40; stu_ptr[2].score = 60;
//Again, you can access the dynamically allocated memory
//by the array-accessing fashion
```

```
stu_ptr = head;
for (i = 0; i < N ; i++)
{
    printf("id: %d with score: %d \n",
           stu_ptr->id, stu_ptr->score);

    stu_ptr++;
}
return 0;
}
```

Topic 2 assignment

- Bubble sort + merge

- Execution result →→→→→

```
Please enter how many numbers in list1: 3
Please input 3 numbers: 7 1 2
Please enter how many numbers in list2: 4
Please input 4 numbers: 2 9 5 4
```

```
sorted list1: 1 2 7
sorted list2: 2 4 5 9
merged list: 1 2 2 4 5 7 9
```

- Use the following structure. In the structure, there is a p_list to store numbers

```
typedef struct num_list
{
    int counts;
    int *p_list;
}tNumList;
```

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  typedef struct num_list
5  {
6      int counts;
7      int *p_list;
8  } tNumList;
9
10 void fill_list(tNumList *list);
11 void bubble_sort(tNumList *list);
12 void merge (tNumList *list1, tNumList *list2);
13 void print_list (tNumList *list);
14
15 int main (void)
16 {
17     tNumList *list1, *list2;
18     list1 = (tNumList *) malloc (sizeof(tNumList));
19     list2 = (tNumList *) malloc (sizeof(tNumList));

```

You should have these four functions

You should have two number lists
→ list1 and list2

- When merge
 - Don't apply the bubble_sort function
 - You only need to traverse list1 and list2, and then print directly
 - No new list is allowed
 - Only list1 and list2 in your program
 - Don't extend the contents of list1 and list2
- You cannot declare any array !
 - But, you can access data by the array-accessing fashion
- The p_list should be dynamically allocated

交作業時 只需要上傳 .c 檔案 Only C file is required when you upload your program

- 若上傳其他不相關的檔案，扣分！！！！
- You will receive a punishment if you upload any other files



| 作業維護 | 題庫維護 | 作業批改

作業資訊 | 挑選題目 | 排列與配分 | 作業預覽

作業名稱 正體中文 week 02 assignment 填寫更多語言 最少要

Deliver C file only and directly !!!

作答說明/師長叮嚀 顯示於