計算機概論與程式設計

LAB 10
2022/12/5
黃奕軒

Recap: called by value

- What is the answer of int a?

```
a = 12
```

```
void call_by_value(int in)
{
    in = 34;
}

int main()
{
    int a = 12;

    call_by_value(a);

    printf("a = %d\n", a);

    return 0;
}
```

Recap: called by reference

- What is the answer of int a?

```
a = 34
```

```
void call_by_ref(int *in)
{
    *in = 34;
}
```

```
int main()
{
    int a = 12;

    call_by_ref(&a);

    printf("a = %d\n", a);

    return 0;
}
```

Struct: called by value

- What is the answer of var1 & var2 of struct a?

```
a.var1 = 12
a.var2 = 34
```

- Note:
  - example_t in => is an instance of struct
    - We should use '.' when we operate the member of in

```c
struct example_t
{
    int var1;
    int var2;
};

void call_by_value(struct example_t in)
{
    in.var1 = 56;
    in.var2 = 78;
}

int main()
{
    struct example_t a;
    a.var1 = 12;
    a.var2 = 34;

    call_by_value(a);

    printf("a.var1 = %d\n", a.var1);
    printf("a.var2 = %d\n", a.var2);

    return 0;
}
```

Struct: called by reference

- What is the answer of var1 & var2 of struct a?

```
a.var1 = 56
a.var2 = 78
```

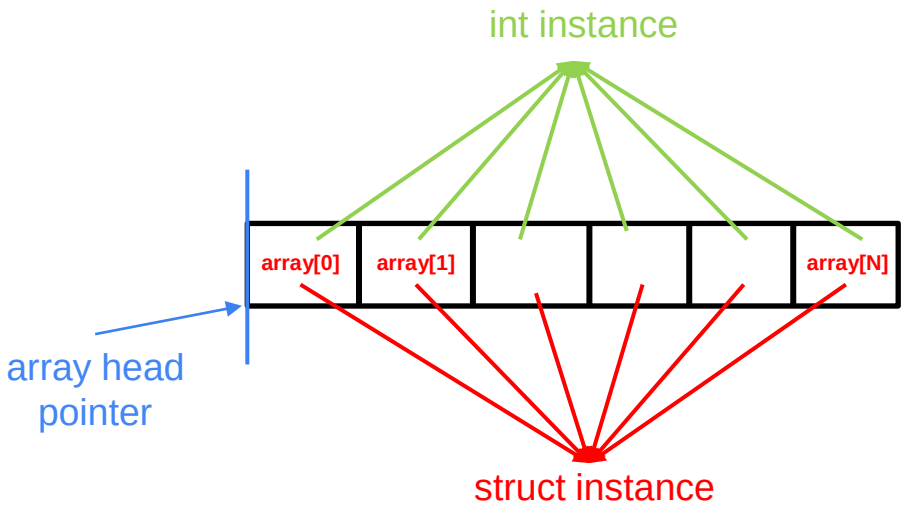
- Note:
  - example_t *in => is a pointer of struct
    - We should use '->' when we operate the member of in

```c
struct example_t
{
    int var1;
    int var2;
};

void call_by_ref(struct example_t *in)
{
    in->var1 = 56;
    in->var2 = 78;
}

int main()
{
    struct example_t a;
    a.var1 = 12;
    a.var2 = 34;

    call_by_ref(&a);

    printf("a.var1 = %d\n", a.var1);
    printf("a.var2 = %d\n", a.var2);

    return 0;
}
```

Struct: array declare

- Recap
  - How to declare int array?
    - int array[100];
  - How to operate int array?
    - array[0] = 1;
    - array[1] = 23;
- Struct
  - How to declare struct array
    - struct example_t array[100];
  - How to operate struct array
    - array[0].var1 = 1;
    - array[0].var2 = 23;
    - array[1].var1 = 4;
    - array[2].var2 = 56;

not ‘->’




```
struct example_t
{
    int var1;
    int var2;
};
```

Lab 10-1 Establish score database

- Given you a score table
  - 4 subject
  - N data (0 < N <= 100)

Sample Input #1

```
5
NAME      STUDENT_ID CHINESE ENGLISH MATH CS
JUSTICE   300123724  72      56      45   67
EMELIA    300123423  12      100     0    45
REAGAN    300123754  23      43      78   88
ADRIANA   300123964  92      47      82   42
RACHEL    300123377  98      75      23   67
```

- Please output the information of each student by input sequence.

The output order of subject is different from input data

Using a structure to store data can easily solve this problem

Sample Output #1

```
---
Name: JUSTICE
Student ID: 300123724
Computer Science: 67
Chinese: 72
Math: 45
English: 56
---
Name: EMELIA
Student ID: 300123423
Computer Science: 45
Chinese: 12
Math: 0
English: 100
---
Name: REAGAN
Student ID: 300123754
Computer Science: 88
Chinese: 23
Math: 78
English: 43
---
Name: ADRIANA
Student ID: 300123964
Computer Science: 42
Chinese: 92
Math: 82
English: 47
---
Name: RACHEL
Student ID: 300123377
Computer Science: 67
Chinese: 98
Math: 23
English: 75
```

Lab 10-2 Sort score by subject

- Given you a score table
  - 4 subject
  - N data (0 < N <= 100)

Sample Input #1

```
5
NAME      STUDENT_ID CHINESE ENGLISH MATH CS
JUSTICE   300123724  72      56      45   67
EMELIA    300123423  12      100     0    45
REAGAN    300123754  23      43      78   88
ADRIANA   300123964  92      47      82   42
RACHEL    300123377  98      75      23   67
```

- Please output the sorted results of each subject in order:
  - Output sorted result by CS
  - Output sorted result by CHINESE
  - Output sorted result by MATH
  - Output sorted result by ENGLISH

Some people might have the same scores in the same subject

You should keep order of input sequence.

You can get hint from OJ

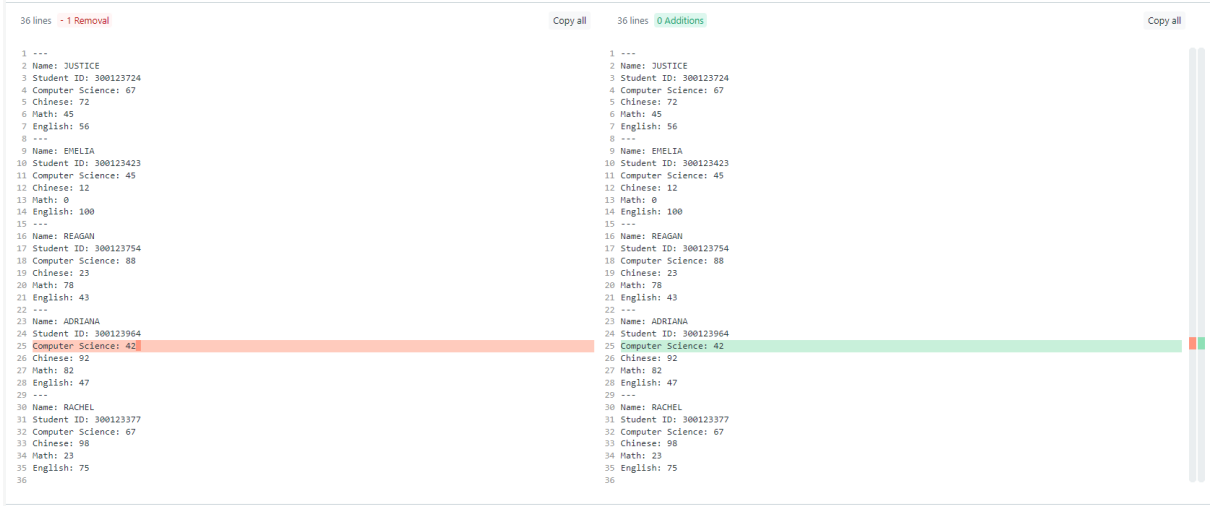
Sample Output #1

```
Sort by Computer Science:
REAGAN, 88
JUSTICE, 67
RACHEL, 67
EMELIA, 45
ADRIANA, 42

Sort by Chinese:
RACHEL, 98
ADRIANA, 92
JUSTICE, 72
REAGAN, 23
EMELIA, 12

Sort by Math:
ADRIANA, 82
REAGAN, 78
JUSTICE, 45
RACHEL, 23
EMELIA, 0

Sort by English:
EMELIA, 100
RACHEL, 75
JUSTICE, 56
ADRIANA, 47
REAGAN, 43
```

Debug

- https://www.diffchecker.com/diff/

```
maxhuang@2080ti:~/ta_lab/lab10_1$ ./main < ../lab10-sample-input.txt
---
Name: JUSTICE
Student ID: 300123724
Computer Science: 67
Chinese: 72
Math: 45
English: 56
---
Name: EMELIA
Student ID: 300123423
Computer Science: 45
Chinese: 12
Math: 0
English: 100
---
Name: REAGAN
Student ID: 300123754
```



Grading

- If you pass TA's demo, you will get lab10-1 & 10-2 score of Formosa OJ
  - Lab10-1 (70%)
  - Lab10-2 (30%)
    - testcase_no_repeat (20%):
      - Everyone scores differently on the same subject
    - testcase_some_repeat (10%):
      - Some people have the same scores in the same subject
      - You should keep order of input sequence

- Total: 100%
  - Formosa OJ:
    - https://oj.nctu.edu.tw/