

Creative Software Design, Assignment 2-2

Handed out: Sep 9, 2021

Due: 23:59, Sep 14, 2021 (NO SCORE for late submissions!)

- Only files submitted by **git push to this course project** at [<Year>_<Course no.>_<Class code>/<Year>_<Course no.>_<Student ID>.git](https://hconnect.hanyang.ac.kr)) will be scored.
- Place your files under the directory structure **<Assignment name>/<Problem no.>/<your files>** just like the following example.

```
+ 2020_ITE0000_2019000001
+ 2-1/
+ 1/
+   - 1.c
+   - Makefile
+ 2/
+   - 2.c
+   - Makefile
+ ...
```

- The submission time is determined not when the commit is made **but when the git push is made**.
 - Your files must be committed to the **master branch**. Otherwise, it will not be scored.
 - Your program should output correct results even for inputs other than those used in the example.
 - Basically, assignments are scored based on the output results. If it is not possible to check whether a requirement is implemented because the output is not correct, no score is given for the requirement, even if it is implemented internally. However, even if the output result is correct, no score is given for a requirement if the internal implementation does not satisfy the requirement.
1. Complete the code skeleton below to write a program that prints out each student's name and his/her corresponding input points.

- A. Take three students and three points per student as an input.
- B. Create three variables of Person type (see the code skeleton below) and fill in the values of each member by taking three points from the user input.
 - i. Assume that each student's name is less than 7 letters in alphabet.
 - ii. Assume that each point is in 2-dimensional orthogonal coordinate system.
 - iii. Take names and its points (x, y) sequentially as the example below.
- C. **Input:** Three sequential inputs of student names less than 7 letters and 6 integers (three 2-dimensional points for each student).
- D. **Output:** Three sequential outputs of student names and his/her points.
- E. Files to submit:
 - i. A C++ source file
 - ii. A Makefile to generate the executable

Code skeleton

```
typedef struct
{
    double x;
    double y;
} Point;

typedef struct
{
    char name[7]
    Point points[3];
} Person;

int main(void){

    //implement this function
```

```
}
```

```
$/touch_points
John 1 2 3 4 5 6
Amy 1 1 -2 -1 0.1 0.1
Emma 1 1 2 2 10 1
John (1 2) (3 4) (5 6)
Amy (1 1) (-2 -1) (0.1 0.1)
Emma (1 1) (2 2) (10 1)
$
```

2. Write a program that prints out each student's name and his/her farthest point from (0,0).
 - A. Append few lines to the previous code.
 - B. Print out each student's name and his/her farthest point from (0,0) as the example below.
 - C. **Input:** Three sequential inputs of student names less than 7 letters and 6 integers (three 2-dimensional points for each student).
 - D. **Output:** Three sequential outputs of student names and his/her farthest point from (0,0).
 - E. Files to submit:
 - i. A C++ source file
 - ii. A Makefile to generate the executable

```
$/touch_points
John 1 2 3 4 5 6
Amy 1 1 -2 -1 0.1 0.1
Emma 1 1 2 2 10 1
John (5, 6)
Amy (-2, -1)
Emma (10, 1)
$
```