User Id : weizhe.goh@digipen.edu | Started : 2020.05.19 19:32:25 | Score : 100%

DigiPen | # Moving From C To C++ | **Assignment**

Rules

Read carefully and check all rules you agree with:

- ☑ Your code must represent your own individual work. Cheating of any kind (copying someone else's work, allowing others to copy your work, collaborating, etc.) will not be tolerated and will be dealt with SEVERELY.

- ☑ Each exercise has description which must be strictly followed.

- ☑ All programs must pass all tests in the main function (when given) to get the final grade. You are not allowed to make any change in the main function in this case.

- ☑ Keep the code proper formatted (correct indentation, max line width is 40 characters).

- ☑ Every week the instructor is available **during the lab time** to discuss following matters:
  - your disagreement with rule in this card,
  - misunderstanding of the current assignment specs,
  - solution for given problems.

Q&A

- Q:... A:You can develop the code using command-line compilers installed on your computer or using online compiler provided by jdoodle.com or ideone.com.

- Q:... A: If you develop your code outside and the output is as expected, copy/paste your code into this service, save it and run. If the output is the same and correct, submit the card.

- Q:... A: Resubmission is possible when your output does not match or you change the code.

Specs

1. Define a struct AABB for axis-aligned bounding boxes that have all sides set parallel to all axes of the coordinate system.

- Define a struct Point for points in the two dimensional space.

- Define the AABB structure with lower left and upper right extreme points using above Point structure.

Develop a function aabb that passes given tests in main function.

- The first parameter of the function is an array of points. The second parameter is the number of points in the array.

- Return the AABB from the function.

- Try to use only one for loop in the function.

★   c   ☑

- Make your code for ANSI C compiler.

Run

```c
#include <stdio.h>

struct Point
{
  int x;
  int y;
};

struct AABB
{
  struct Point min;
  struct Point max;
};

struct AABB aabb(struct Point pp[],
                             int size)
{
  struct AABB aabb;
  int i = 0;

  aabb.min.x = pp[0].x;
  aabb.max.x = pp[0].x;
  aabb.min.y = pp[0].y;
  aabb.max.y = pp[0].y;

  for(i=1;i<size;i++)
  {
    if(pp[i].x <= aabb.min.x)
```

```
                {
                    aabb.min.x = pp[i].x;
                }

                if(pp[i].x >= aabb.max.x)
                {
                    aabb.max.x = pp[i].x;
                }

                if(pp[i].y <= aabb.min.y)
                {
                    aabb.min.y = pp[i].y;
                }

                if(pp[i].y >= aabb.max.y)
                {
                    aabb.max.y = pp[i].y;
                }
        }
        return aabb;
}
```

```c
// Do not edit below this line
int main()
{
    struct Point pp[] =
        {{1,4},{-1,8},{4,-7},{0,3}};
    struct AABB value = aabb(pp,4);
    printf("(%d,%d)(%d,%d)",
        value.min.x,value.min.y,
        value.max.x,value.max.y);
    return 0;
}
```

```
(-1,-7)(4,8)
```

★ c++ ☑

- Change your previous code for C (including main function) using C++ differences. Make sure that output is exactly the same.

Run

```cpp
#include <iostream>

struct Point
{
    int x;
    int y;
};

struct AABB
{
    Point min;
    Point max;
};

AABB aabb(Point pp[], int size)
{
    AABB aabb;
```

Survey

- What is approximate number of hours you spent implementing this assignment?

2hrs

- Indicate the specific portions of the assignment that gave you the most trouble

Conversion of C++, not being taught fully on the conversion

```cpp
    AABB aabb;

    aabb.min.x = pp[0].x;
    aabb.max.x = pp[0].x;
    aabb.min.y = pp[0].y;
    aabb.max.y = pp[0].y;

    for(int i=1;i<size;i++)
    {
      if(pp[i].x <= aabb.min.x)
      {
        aabb.min.x = pp[i].x;
      }

      if(pp[i].x >= aabb.max.x)
      {
        aabb.max.x = pp[i].x;
      }

      if(pp[i].y <= aabb.min.y)
      {
        aabb.min.y = pp[i].y;
      }

      if(pp[i].y >= aabb.max.y)
      {
        aabb.max.y = pp[i].y;
      }
    }
    return aabb;
  }

  // Do not edit below this line
  int main()
  {
    Point pp[] =
          {{1,4},{-1,8},{4,-7},{0,3}};
    AABB value = aabb(pp,4);

    std::cout << "(" << value.min.x <<
","
    << value.min.y << ")";

    std::cout << "(" << value.max.x <<
","
    << value.max.y << ")";
    return 0;
  }
```

(-1,-7)(4,8)