Object-Oriented Programming Language

03/07/2024

Due 3/07 11:59AM

Lab Assignment 3

Please take a look at **file_io.cpp** and **comma.cpp** on the course webpage in lecture 3 for more examples on how file I/O work. There is also the stolf(), atof() and atoi() functions that can convert strings to `float`s and `int`s[1].

1. Write a program that inputs 3 triangles each with three vertices. Store them in an **array**, and then determine if the given triangles are right triangles (直角三角形). Here are some sample code you can use.

```
struct Vertex {
double x;
double y;
};
```

```
struct Triangle {
Vertex A;
Vertex B;
Vertex C;
bool right;
};
```

A sample run of the program is as follows:

[1] https://www.geeksforgeeks.org/cpp-program-for-string-to-double-conversion/
https://www.geeksforgeeks.org/convert-string-to-int-in-cpp/

```
Input for the 1 triangle:
        Please input the x & y coordinates of the 1st vertex: 0 3
        Please input the x & y coordinates of the 2nd vertex: 4 0
        Please input the x & y coordinates of the 3rd vertex: 0 0
Input for the 2 triangle:
        Please input the x & y coordinates of the 1st vertex: 4 4
        Please input the x & y coordinates of the 2nd vertex: 4 4
        Please input the x & y coordinates of the 3rd vertex: 3 3
Input for the 3 triangle:
        Please input the x & y coordinates of the 1st vertex: 9 7
        Please input the x & y coordinates of the 2nd vertex: 4 5
        Please input the x & y coordinates of the 3rd vertex: 12 -5
Done reading Triangles.
Triangle 1 is a right triangle!
The right angle of triangle 1 is at the Vertex3
Triangle 2 is NOT a right triangle!
Triangle 3 is NOT a right triangle!
```

2a. Write a C++ program that inputs certain number of triangles each with three vertices. Store them in a **vector**, and then determine if the given triangles are right triangles (直角三角形). When you access the triangles in the vector, use **iterators**. You should also use the mentioned **struct** in problem 1.

```
How many Triangles will you input? 3
Input for the 1 triangle:
        Please input the x & y coordinates of the 1st vertex: 1 1
        Please input the x & y coordinates of the 2nd vertex: 1 -1
        Please input the x & y coordinates of the 3rd vertex: 0 0
Input for the 2 triangle:
        Please input the x & y coordinates of the 1st vertex: 1 1
        Please input the x & y coordinates of the 2nd vertex: 2 2
        Please input the x & y coordinates of the 3rd vertex: 0 1
Input for the 3 triangle:
        Please input the x & y coordinates of the 1st vertex: 1 1
        Please input the x & y coordinates of the 2nd vertex: 1 1
        Please input the x & y coordinates of the 3rd vertex: 1 1
Done reading Triangles.
Triangle 1 is a right triangle!
Triangle 2 is NOT a right triangle!
Triangle 3 is NOT a right triangle!
```

```
How many Triangles will you input? 2
Input for the 1 triangle:
        Please input the x & y coordinates of the 1st vertex: 1 1
        Please input the x & y coordinates of the 2nd vertex: 1 -1
        Please input the x & y coordinates of the 3rd vertex: 0 0
Input for the 2 triangle:
        Please input the x & y coordinates of the 1st vertex: 1 1
        Please input the x & y coordinates of the 2nd vertex: 1 1
        Please input the x & y coordinates of the 3rd vertex: 1 1
Done reading Triangles.
Triangle 1 is a right triangle!
Triangle 2 is NOT a right triangle!
```

3a. Write a program that takes a (r x c) matrix as input and return the transpose of the matrix.

r,c <=10

A sample run of the program is as follows:

```
2 3

4 5 6
3 4 5

4 3
5 4
6 5
```

3b. Write a program that takes a (r x c) matrix as input from matrix.txt and print the transpose of the matrix in matrix_out.txt.

Input text in matrix.txt:

```
2 3
4 5 6
3 4 5
```

Output in matrix_out.txt:

```
4 3
5 4
6 5
```

Input text in matrix.txt:

```
3 4
4 5 6 7
1 2 3 4
0 0 0 0
```

Output in matrix_out.txt:

```
4 1 0
5 2 0
6 3 0
7 4 0
```

4. Write a program that input a string and test if it is a palindrome(回文)
You **can't** use the reverse() in the #include <algorithm>
(hint: input N character, compare the first and the N-th, second and the (N-1)-th and so on)
Some sample runs of the program are as follows:

```
abcba
yes
```

```
wwww
yes
```

```
abbbnnncaca
no
```

5. Write a program that input a number as a **string** and return its reverse
(hint: You can use reverse() in the #include <algorithm> and there is no '0' at the beginning but the end of the input number)
Some sample runs of the program are as follows:

```
76355200
255367
```

```
12345
54321
```

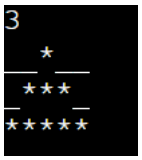
6. Write a program that input an arithmetic sequence's (等差數列) first number, last number and the common difference of successive members. Please return the whole arithmetic sequence.
Some sample runs of the program are as follows:

```
3 9 2
3 5 7 9
```

```
1 17 4
1 5 9 13 17
```

7. Write a program that input a number n and return a “cool” pyramid. For example, if n=3, first row has 1 * and 4_, second row have 3* and 2_, third row has only 5*. That is, creating a (2n-1) x(2n-1) matrix.

Some sample runs of the program are as follows:



8. There is very simple encryption method in cryptography(密碼學) which is add a certain integer K to each character of the code to obtain the characters of the password For example, if K=2, then apple becomes crrng after encryption .Write a program that input a string and return a string after **encryption.**

Please use K=7

Some sample runs of the program are as follows:

```
I love OOP, OOP NO:1
P'sv}l'VVW3'VVW'UVA8
IBM is a trademark of the International Business Machine Corporation.
PIT'pz'h'{yhklthyr'vm'{ol'Pu{lyuh{pvuhs'I|zpulzz'Thjopul'Jvywvyh{pvu5
```

You can copy the sample input:

I love OOP, OOP NO:1

IBM is a trademark of the International Business Machine Corporation.

9. Please calculate the average score from the score.txt file which separates the scores with commas.

```
Please input file name: scores.txt
scores read: 59.2 81.3 66.2 99 100 75.3 88.8
There are 7 scores and the average is 81.4
```

```
Please input file name: scores2.txt
scores read: 59.2 73 81.3 66.2 92.1 99 100 62.9 75.3 88.8 97.6
There are 11 scores and the average is 81.4
```