**CPP Problem Design**

|  |
| --- |
| **Subject: Simple Drawing Program** |
| **Contributor: 謝宜杭,** **林承達, 廖宣瑋** |
| **Main testing concept: 2-Dimension Array**   |  |  | | --- | --- | | **Basics** | **Functions** | | □ C++ BASICS  □ FLOW OF CONTROL  □ FUNCTION BASICS  □ PARAMETERS AND OVERLOADING  ■ ARRAYS  □ STRUCTURES AND CLASSES  □ CONSTRUCTORS AND OTHER TOOLS  □ OPERATOR OVERLOADING, FRIENDS,AND REFERENCES  □ STRINGS  □ POINTERS AND DYNAMIC ARRAYS | □ SEPARATE COMPILATION AND NAMESPACES  □ STREAMS AND FILE I/O  □ RECURSION  □ INHERITANCE  □ POLYMORPHISM AND VIRTUAL FUNCTIONS  □ TEMPLATES  □ LINKED DATA STRUCTURES  □ EXCEPTION HANDLING  □ STANDARD TEMPLATE LIBRARY  □ PATTERNS AND UML | |
| **Description:**  Please design a simple drawing program (fill the background with \*) allow users to draw square, isosceles right triangle and lines on the console. (for figures, fill with Upper X)  **Input:**  In the beginning, allow users to enter two integers separated by space to set the size of the drawing board (**m**\***n**). After that, enter the corresponding drawing code. First input the type of the figure, **S** indicates square, **T** indicates Isosceles right triangle and **L** indicates lines. Finally, input the parameter of the figure. Following is the required parameters of each type of figure:   * **S** **<w> <x> <y>**   Note that the integer **w** is the width and two integers (**x**, **y**) is the origin of the square. The direction of the square extends from top to bottom, from left to right.   * **T <w> <x> <y> <LU/LD/RU/RD>**   Integer **w** is the length of hypotenuse. (**x**, **y**) is the origin of the triangle. **LU** (Left Up), **LD** (Left Down), **RU** (Right Up) and **RD** (Right Down) indicates the side that the triangle faces. Draw an equilateral triangle with side length **w** from point (**x**, **y**) that facing the specified direction.   * **L** **<x1> <y1> <x2> <y2>**   Input two pairs of integers separated by space, (**x1**, **y1**) and (**x2**, **y2**). Draw a line segment with (**x1**, **y1**) and (**x2**, **y2**) as the endpoints.   * **EXIT**   Exit the program.    ※ The origin point (0, 0) of the drawing board is left up.  ※ If the figure to be drawn exceeds the boundary of the drawing board, print "**Out of range**".  ※ For any figure, if the given width is **1**, you need to draw the origin point.  ※ Output the drawing board after each command and separates each with an empty line.  ※ We make sure that all the input data is drawable.  **Output:**  After drawing a figure or an error message, output a newline.  **Sample Input / Output：**   |  |  | | --- | --- | | Sample Input | Sample Output | | 5 6  S 2 0 0  S 2 100 100  L 0 4 4 4  T 2 1 3 LU  EXIT | XX\*\*\*  XX\*\*\*  \*\*\*\*\*  \*\*\*\*\*  \*\*\*\*\*  \*\*\*\*\*  Out of range.  XX\*\*\*  XX\*\*\*  \*\*\*\*\*  \*\*\*\*\*  XXXXX  \*\*\*\*\*  XX\*\*\*  XX\*\*\*  \*X\*\*\*  XX\*\*\*  XXXXX  \*\*\*\*\* | |
| **□ Eazy,Only basic programming syntax and structure are required.**  **■ Medium,Multiple programming grammars and structures are required.**  **□ Hard,Need to use multiple program structures or more complex data types.** |
| **Expected solving time:**  30 minutes |
| **Other notes:** |