Version: 04/10/2021 21:46

CSC1016S Assignment 7

Polymorphism

Assignment Instructions

This assignment involves constructing programs in Java using sub-classes, inheritance and polymorphism.

Question 1 [40 marks]

Write a program called `Question1.java` to store some details of computer boxes (serial number, manufacturer, colour, amount of memory), screens (serial number, manufacturer, colour, size) and accessories (serial number, manufacturer, colour) in a single array, with the options of listing all parts or deleting a particular part.

Your program must continuously prompt the user to choose an option from a list and act on that option, until the exit option is chosen. See the sample output below. If a part is not found, output "Not found" instead of "Done".

You must use inheritance and polymorphism to model your Box, Screen and Accessories classes (use those exact class names) as subclasses of the same base class, which forms the basis for the array.

Note: Be very careful to reproduce the output exactly. Copy-and-paste is highly recommended to avoid minor typographical errors that drive you crazy when submitting online!

Sample IO

```
Welcome to Great International Technology
MENU: add (B)ox, add (S)creen, add (A)ccessories, (D)elete, (L)ist, (Q)uit
b
Enter the serial number
123
Enter the manufacturer
Lenovo
Enter the colour
Beige
Enter the amount of memory (MB)
2048
Done
MENU: add (B)ox, add (S)creen, add (A)ccessories, (D)elete, (L)ist, (Q)uit
s
Enter the serial number
456
Enter the manufacturer
Phillips
Enter the colour
Beige
Enter the screen size in inches
19
```

```
Done
MENU: add (B)ox, add (S)creen, add (A)ccessories, (D)elete, (L)ist, (Q)uit
a
Enter the serial number
789
Enter the manufacturer
Logitech
Enter the colour
Beige
Done
MENU: add (B)ox, add (S)creen, add (A)ccessories, (D)elete, (L)ist, (Q)uit
d
Enter the serial number
456
Done
MENU: add (B)ox, add (S)creen, add (A)ccessories, (D)elete, (L)ist, (Q)uit
l
Box: 123, Lenovo, Beige, 2048
Accessories: 789, Logitech, Beige
Done
MENU: add (B)ox, add (S)creen, add (A)ccessories, (D)elete, (L)ist, (Q)uit
q
```

Question 2 [60 marks]

Write a program to manage a set of graphical objects based on the commands in a specified file. This is a graphical equivalent of the famous **edlin** tool for text manipulation.

On the Vula page of the assignment, you will find VectorGraphics and Question2 classes that will handle the file I/O for you. When you run Question2, specify the file name like this:

```
java Question2 myfile.in
```

Each line of the file contains an instruction in one of the formats given below:

```
a <id> <x> <y> rectangle <x_length> <y_length>
a <id> <x> <y> hline <x_length>
a <id> <x> <y> vline <y_length>
a <id> <x> <y> ptline <x1> <y1>
w
d <id>
m <id> <x> <y>
x
```

Explanation:

'a' adds an object of a particular type (rectangle/hline/vline/ptline) with the specified parameters.

'w' renders/draws the objects and prints the composite image to the screen.

'd' deletes an object based on its id.

'm' moves an object to a new position based on its id.

‘x’ exits the program.

Note:

- A 'ptline' differs from the other objects in that the values after the object type are not lengths, but instead a second point described by *x1* and *y1*. This object represents a line from *(x, y)* to *(x1, y1)*.
- The answers the automatic marker expects were generated using an implementation of Bresenham's Line Drawing Algorithm. (See the appendix).
- Assume that the drawing canvas is 20 blocks wide (in the *x* direction) and 20 blocks high in the *y* direction), starting with (0,0) in the top left.

As mentioned before, you are given class that will serve as the framework for this program on the Vula page, including the main class that does the input and output. You are only required to create the **Rectangle**, **HLine**, **VLine** and **PtLine** subclasses of the given **VectorObject** base class. You should add comments to all classes you write.

Sample Input File

```
a 1 5 5 rectangle 10 10
a 2 3 3 hline 14
a 3 3 16 hline 14
a 4 3 3 vline 14
a 5 16 3 vline 14
w
d 1
m 2 3 5
m 3 3 14
m 4 5 3
m 5 14 3
a 6 6 6 ptline 13 13
a 6 13 6 ptline 6 13
w
x
```

Sample Output

```
+--------------------+
+                    +
+                    +
+                    +
+   **************   +
+   *            *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   * ********** *   +
+   *            *   +
+   **************   +
+                    +
+                    +
+                    +
+--------------------+
+--------------------+
+                    +
+                    +
+                    +
+     *        *     +
+     *        *     +
+   **************   +
+     **      **     +
+     * *    * *     +
+     *  *  *  *     +
+     *   **   *     +
+     *   **   *     +
+     *  *  *  *     +
+     * *    * *     +
+     **      **     +
+   **************   +
+     *        *     +
+     *        *     +
+                    +
+                    +
+                    +
+--------------------+
```

Marking and Submission

Submit your assignment as a set of Java files named **Question1.java, Rectangle.java**, **HLine.java**, **VLine.java**, **PtLine.java** containing classes with corresponding names, as well any other Java files you have created, in a Zip file.

Appendix

Pseudocode for line drawing algorithm

```
def draw($x_0$, $y_0$, $x_1$, $y_1$)=
    steep = $|y_1-y_0|>| x_1-x_0|$
    if steep then
        swap($x_0$, $y_0$)
        swap($x_1$, $y_1$)
    end if
    if $x_0>x_1$ then
        swap($x_0$, $x_1$)
        swap($y_0$, $y_1$)
    end if
    if y0<y1 then
        ys = 1
    else
        ys = -1
    m = $|y_1-y_0|/(x_1-x_0)$
    y = $y_0$
    error = 0

    for x = x0 to x1 do
        if steep then
            plot(x, y)
        else
            plot(y, x)
        end if
        error = error + m
        if error>0.5 then
            y = y+ys
            error = error-1
        end if
    end for
end
```

NOTE:

- This code does not deal with horizontal or vertical lines.
- $|a|$ represents the absolute value of *a* i.e. the value without regard to its sign.