CS141 programming II (java)

Assignment 2

Chapter 12 – OOD

Chapter 13 - Recursion

Assignment 2 worth 5 points, Due on Sunday 17/10/2016. Will deduct 1 point for each day late. Copying each other work will result in giving 0 to both.

Q1: Briefly define the term Waterfall model? And list the phases of the development process in order they are done? (5 points)

Answer:

Waterfall Is a software development method where each phase is done in sequential order: analysis, design, implementation, testing, and deployment. Where you cannot start a phase until the phase before it is completely done.

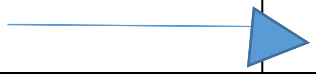
- *Analysis*
- *Design*
- *Implementation*
- *Testing*
- *Deployment*

Q2: Fill in the table below by drawing the correct arrow that represent the relationship between classes? And Indicate Type of relationship: if it is an "IS-A" or "HAS-A", or "USES" relationship if applicable(5 points)

Relationship	Draw	Type of relationship
Inheritance		

Interface Implementation		
Aggregation		
Dependency		

Answer:

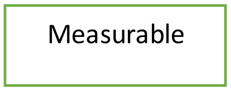
Relationship	**Line Style**	**Arrow Tip**	**Draw**	**Type of relationship**
Inheritance	Solid	Triangle		IS-A
Interface Implementation	Dotted	Triangle		USES
Aggregation	Solid	Diamond		HAS-A
Dependency	Dotted	Open		USES

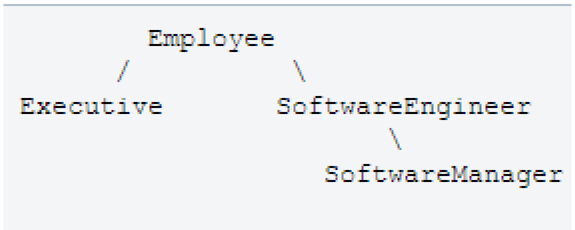
Q3: Given Question 5 from the previous Assignment1. (10 points)

Draw the correct UML diagram to represent the relationships between all classes.

Add an Interface called Measurable and indicate in the diagram that the Employee class implements Measurable interface.

For each class in your diagram do the CRC card for each to indicate responsibilities and collaborators.





```
         Employee
       /          \
Executive        SoftwareEngineer
                        \
                    SoftwareManager
```

Answer: (don't have time to draw a full diagram but the answer as follows)

UML Drawing,

The Inheritance arrow is a solid line with closed triangle arrow pointing from the child to the parent

The interface arrow is a dotted line with open arrow pointing in this case from Employee to Measureable

CRC

Class Employee

work()

toString()

Collaborators

Measurable

And so on for the rest of the classes

The subclass will have the superclass as the collaborator.

Q4: Write and run the java program that computes the Factorial of a given number using a Recursive method. (20 points)

Submit your program files as .java file that can be run. And submit in the document sample run for example given the value 4 the factorial is 24

Q4: Write and run the java program that computes the Factorial of a given number using a Recursive method. (20 points)

Submit your program files as .java file that can be run. And submit in the document sample run for example given the value 4 the factorial is 24

Solution:

```
/*
        Java Factorial Using Recursion Example
        This Java example shows how to generate factorial of a given number
        using recursive function.
*/

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class JavaFactorialUsingRecursion {

        public static void main(String args[]) throwsNumberFormatExceptio
n, IOException{

                System.out.println("Enter the number: ");

                //get input from the user
BufferedReader br=new BufferedReader(newInputStreamReader(System.in
));
                int a = Integer.parseInt(br.readLine());

                //call the recursive function to generate factorial
                int result= fact(a);


                System.out.println("Factorial of the number is: " +result);
        }

        static int fact(int b)
        {
                if(b <= 1)
                        //if the number is 1 then return 1
                        return 1;
                else
                        //else call the same function with the value - 1
                        return b * fact(b-1);
        }
```

```
  }
```

Q5. What is the advantage and disadvantage of using recursion? (5 points)

Answer:

Advantages: recursion is a better solution to many situations that occurs in solving many problems. The code will be simpler, shorter and more elegant.

Disadvantages: recursion performance can be slow, since the method call is repeated and the Operating System will stack the process calls until it is eventually resolved.

Q6. Briefly explain what is infinite recursion, and what causes it to happen? (5 points)

Answer:

Infinite recursion happens when the method keeps calling itself infinitely without an end.

It could happen when the code is incorrect, that is there is no terminating or base condition. Example of a correct terminating conition. if (width == 1) { return 1; }

Or if the recursive call to the method is sent without simplified parameter. Example of correct recursive call with reduced parameter. factorial(n-1)

Terminating condition:

if (n == 1) { return; } // that is the terminating condition without it we will have infinite recursion.

Recursive statement:

fact (n -1) // the method fact() calls itself with reduced parameter.