Please fill in your Student Number

Student Number : _____

Name:

_____

_____

Student Number:

_____

University of Cape Town ~ Department of Computer Science

Computer Science 1016S ~ 2009

Test 2

Question	Max	Mark	Internal	External
1	12			
2	12			
3	6			
TOTAL	**30**			

Marks : 30
Time : 40 minutes
Instructions:
a) Answer all questions.
b) Write your answers in the space provided.
c) Show all calculations where applicable.

Question 1: Sorting [12 marks]

Use the following code snippet to answer question 1:

```
1  public class ASorter {
2       public void sort(double[] a, int begin, int end) {
3          if ((end - begin) >= 1) {
4              int splitPoint = split(a, begin, end);
5              sort(a, begin, splitPoint);
6              sort(a, splitPoint + 1, end);
7              join(a, begin, splitPoint, end);
8          }
9      }
10
11     protected int split(double[] a, int begin, int end)
12     {
13         double[] temp;
14         int size = (end - begin + 1);
15         temp = new double[size];
16
17         double splitValue = a[begin];
18         int up = 0;
19         int down = size - 1;
20
21         for (int i = begin + 1; i <= end; i++)
22         {
23             if (a[i] <= splitValue)
24             {
25                 temp[up] = a[i];
26                 up++;
27             }
28             else
29             {
30                 temp[down] = a[i];
31                 down--;
32             }
33         }
34
35         temp[up] = a[begin];
36
37         for (int i = 0; i < size; i++)
38             a[begin + i] = temp[i];
39
40         return (begin + up);
41     }
42
43     protected void join(double[] a, int begin, int splitPoint,
                                int end)
44     {
45         //Nothing to do.
46     }
47 }
```

a) What kind of sorting algorithm is it? [2]

b) What is the complexity of the sort (i.e. on average, how many comparisons does the sort need to arrange an array of N elements?) [2]

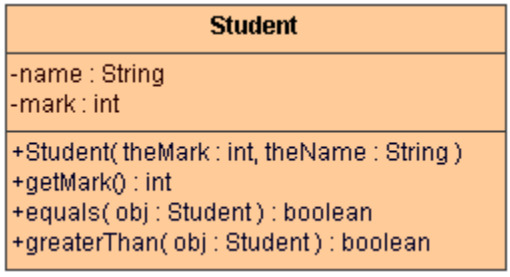
c) Explain what a Comparable interface is and what it is used for. [2]

d) List at least 2 main differences between interfaces and abstract classes. [2]

e) How would you change the above program so that it can sort an array of objects of a class that implements Comparable interface? (i.e. State the steps then name the lines of code that need to be changed?) [4]

Question 2: UML and Design Patterns [12 marks]

Use the following UML diagram to answer question 2:

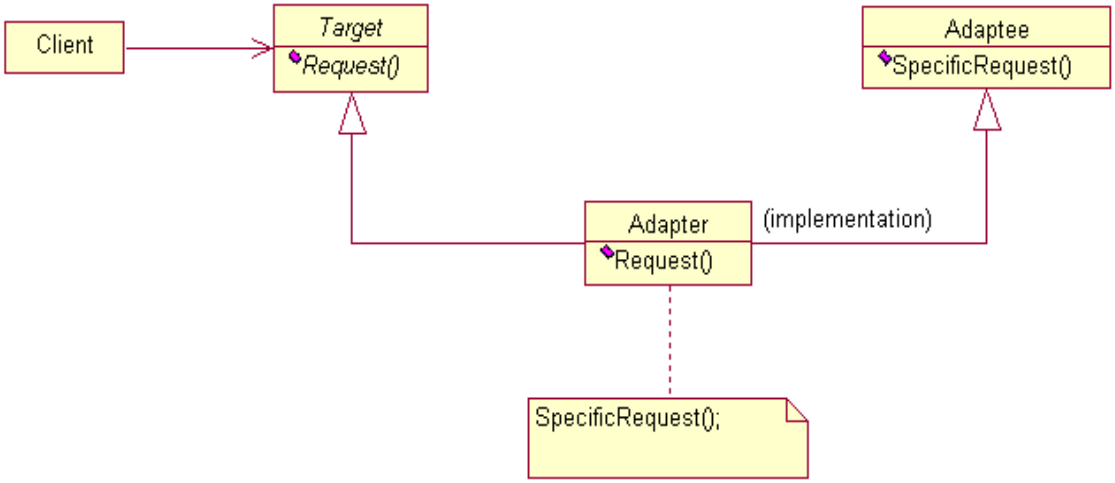


a) Decribe 3 sections of the above class diagram and accessibilities of the class members. [2, 2]

b) Define what a design pattern is and describe a motivation of using design patterns [4]

c) Using the following design pattern, Adaptor pattern, write an adapter class called Student_Comparable so that Student objects can be adapted and sorted using the sorting algorithm in Question 1.d [4]

Note: sorting in Question 1.d using Comparable interface and Student class does not implement Comparable interface



Question 3 [6 marks]

Consider the following partial definition of a simple linked list

```
public class LinkedList<T>
{

    private class Node<T>
    {
        private T data;
        private Node<T> link;

        public Node( )
        {
             data = null;
             link = null;
        }

        public Node(T newData, Node<T> linkValue)
        {
           data = newData;
           link = linkValue;
        }
     }//End of Node<T> inner class


    private Node<T> head;

    public LinkedList ( )
    {
        head = null;
    }

    public void add (T itemData)
    {
        // Fill in
    }


    public void clear( )
    {
        //Fill in
    }
```

a) Fill in the definition of the add method so that items are always added to the head of the list. [3]

b) Fill in the definition for the clear method which empties the list. [1]

c) After clearing the linked list, do you need to remove the nodes from memory? Justify your answer. [1]

_____

_____

_____

_____

_____

_____

_____

_____

d) State one benefit of making the Node class a **private inner class** of LinkedList. [1]

_____

_____

_____

_____

_____

_____