CS1020 Data Structures and Algorithms I

ANSWER SHEETS

Answers included

INSTRUCTIONS TO CANDIDATES

1. This document consists of **SIX (6)** printed pages.
2. Fill in your Matriculation Number clearly below and at the top of pages 3 and 5.
3. Fill in your Tutorial Group below.
4. You may use pencil to write your code.
5. The last page (page 6) may be used if you need more space to write your answers.

MATRICULATION NO.: A

(Write your Matriculation Number legibly with a pen.)

TUTORIAL GROUP:

For examiners' use only		
Question	***Max***	***Marks***
Q1-5	5	
Q6	2	
Q7	3	
Q8	3	
Q9	3	
Q10	5	
Q11	9	
Total	**30**	

MCQs [5 marks]

Q1. C **Q2.** E **Q3.** B **Q4.** C **Q5.** D

Q6. [2 marks]

```
0
Exception in thread "main" java.lang.NullPointerException
        at Q6.main(Q6.java:8)
```

Q7. [3 marks]

```
3 11 39
```

Q8. [3 marks]

```
125 123 125 124
```

Q9. [3 marks]

```
private static Boolean areAnagrams(String word1,
                                   String word2) {
   if (word1.length() != word2.length())
      return false;
   for (int i = 0; i < word1.length(); i++) {
      word2 =
       word2.replaceFirst(word1.substring(i,i+1),"");
```

or

```
      String letter = word1.substring(i, i+1);
      if (word2.contains(letter))
         word2 = word2.replaceFirst(letter, "");
      else
         return false;
```

or other alternatives.

```
   }
   return (word2.length() == 0);
}
```

Q10. Triangle.java **[Total: 5 marks]**

(a) [1 mark]

```
// Default triangle with vertices (0,0), (0,1) and (1,0)
public Triangle() {
  this(0, 0, 0, 1, 1, 0);
 or
  this(new Point(), new Point(0,1), new Point(1,0));
}
```

(b) [1 mark]

```
public Triangle(int vertex1X, int vertex1Y,
                int vertex2X, int vertex2Y,
                int vertex3X, int vertex3Y) {

   this(new Point(vertex1X, vertex1Y),
        new Point(vertex2X, vertex2Y),
        new Point(vertex3X, vertex3Y));

}
```

(c) [3 marks]

```
// Return the area of this triangle
private double computeArea() {

 int minX = Math.min(vertices[0].x,
            Math.min(vertices[1].x, vertices[2].x));

 int maxX = Math.max(vertices[0].x,
            Math.max(vertices[1].x, vertices[2].x));

 int minY = Math.min(vertices[0].y,
            Math.min(vertices[1].y, vertices[2].y));

 int maxY = Math.max(vertices[0].y,
            Math.max(vertices[1].y, vertices[2].y));

 return (maxX – minX)* (maxY – minY) / 2.0;

}
```

Q11. TestTriangle.java **[Total: 9 marks]**

(a) [5 marks]

```
private void createList() {

   Scanner sc = new Scanner(System.in);
   System.out.print("Enter number of triangles: ");
   int num = sc.nextInt();

   triangles = new ArrayList<Triangle>();

   for (int i=0; i<num; i++) {
      System.out.print("Enter 3 vertices: ");

      triangles.add(
         new Triangle(sc.nextInt(),sc.nextInt(),
                      sc.nextInt(),sc.nextInt(),
                      sc.nextInt(),sc.nextInt()));
   }

}
```

Q11. TestTriangle.java

(b) [4 marks]

```
private void removeSmallestTriangle() {


  int minAreaIndex = 0;

  double minArea = triangles.get(0).getArea();

  for (int i=1; i<triangles.size(); i++) {
     if (triangles.get(i).getArea()
          < triangles.get(minAreaIndex).getArea()) {
        minAreaIndex = i;
     }
  }

  triangles.remove(minAreaIndex);
















}
```

This page is intentionally left blank.
Do **NOT** use it for your rough work.
Use it ONLY if you need extra space for your answer, in which case please indicate the **question number clearly**.

— END —