

2004 Paper 10 Question 3

Foundations of Programming

A Java program is being developed to assist with the processing of examination marks. A test program begins as follows:

```
public class Exam
{ private static Mark[] question = {new Mark(8),
                                     new Mark(),
                                     new Mark(6),
                                     ...
```

The program makes use of a `class Mark` which begins:

```
class Mark
{ public boolean attempted;
  public int score;
  ...
```

The `Mark` array `question` has one entry for each candidate so the length of the array indicates the total number of candidates.

An entry such as `new Mark(8)` sets up a `Mark` object whose data field `attempted` is set to `true` and whose data field `score` is set to 8, indicating that the candidate attempted the question and was awarded 8 marks. It may be assumed that every score is an integer in the range 0 to 10 inclusive.

An entry such as `new Mark()` sets up a `Mark` object whose data field `attempted` is set to `false` (and whose data field `score` is arbitrary) indicating that the candidate did not attempt the question.

(a) Supply suitable constructors for `class Mark` [4 marks]

(b) Write two methods `int getCount()` and `double getMean()` which, when handed the actual argument `question`, return the number of candidates who attempted the question and the mean mark achieved by those candidates respectively. If no candidates attempted the question, `getMean()` should return `-1d`. [9 marks]

(c) Write a method `int[] getRank()` which begins:

```
private static int[] getRank(Mark[] q)
{ int[] rank = {0,0,0,0,0,0,0,0,0,0,0};
```

This method should return the `int` array `rank` with each element `rank[i]` set to the number of candidates who scored more than i marks. Note that if the maximum `score` is 9 then both `rank[9]` and `rank[10]` will be zero and `rank[8]` will be the number of candidates who scored 9. [7 marks]