



88137014



**COMPUTER SCIENCE
STANDARD LEVEL
PAPER 2**

Friday 15 November 2013 (morning)

1 hour 30 minutes

INSTRUCTIONS TO CANDIDATES

- Do not open this examination paper until instructed to do so.
- Answer all the questions.
- A clean copy of the **Computer Science** case study is required for this paper.
- The maximum mark for this examination paper is [70 marks].

Answer **all** the questions.

1. Consider the array `data` and the program fragment shown below.

	[0]	[1]	[2]	[3]	[4]	[5]
data	-2	2	4	3	0	-1

```
public double[] doIt(double[] inArray)
{
    int n = inArray.length;
    double[] outArray = new double[n];

    for(int i=1; i<n-1; i=i+1)
    {
        outArray[i] = inArray[i-1] + inArray[i] + inArray[i+1];
        outArray[i] = outArray[i] / 3.0;
    }

    return outArray;
}
```

The method `doIt` is called by the statement `double[] d = doIt(data);`.

(a) By copying and completing the following table, trace the method `doIt` for this call and show the contents of the array `outArray` after each pass through the loop. [4 marks]

outArray	[0]	[1]	[2]	[3]	[4]	[5]
Original	0	0	0	0	0	0
i=1						
i=2						

(b) State the effect of the `doIt()` method on the values in the array `data`. [1 mark]

(c) Outline why the loop cannot start and end on the indices of the first and last elements of the input array. [2 marks]

(d) Construct statements to modify the `doIt()` method so that the first and last elements of the output array are also computed. The first element of the output array will be the average of the first two elements of the input array. The last element of the output array will be the average of the last two elements of the input array. [2 marks]

(This question continues on the following page)

(Question 1 continued)

Similarly, for the two-dimensional array `data2` shown below, an average value can be calculated for each interior element as follows.

The average value for the interior element marked **X** is the average of itself and its four neighbouring elements which are marked **A**, **B**, **C**, and **D**. (Average = $(X+A+B+C+D)/5.0$)

<code>data2</code>	[0]	[1]	[2]	[3]	[4]	[5]
[0]						
[1]		A				
[2]	D	X	B			
[3]		C				
[4]						

(e) Construct a code fragment that computes the average for the value at `data2[1][4]`.

[2 marks]

(f) The method `doIt2(double[][] in2Array)` fills a two-dimensional output array `out2Array[][]` so that:

- each interior element of the array `out2Array` is the average of the corresponding element and its four neighbouring elements (as explained above) in the array `in2Array`
- all elements on the edges of `out2Array` are kept at their default value of zero.

Construct the method `doIt2()`, which is called by the statement `double[][] e = doIt2(data2)`. It has been started below.

```
public double[][] doIt2(double[][] in2Array)
{
    . . .
}
```

[6 marks]

(g) Outline how the elements on the edge of `out2Array` could be given values equal to the average of the corresponding element and its neighbouring elements in `in2Array`.

[3 marks]

2. A program is written to help farm management. The following class is used to create objects representing fields in a farm.

```
public class FarmField
{
    String fieldName;        // Unique name for the field
    int fieldSize;           // Size of the field in square metres
    int soilType;            // A value indicating soil acidity
    int fertilizerType;      // A value indicating the type of fertilizer
    int cropType;           // Crop type: 1=Corn, 2=Soybeans, 3=Alfalfa, etc.
}
```

- (a) (i) Construct an empty `FarmField` object named `firstField`. *[1 mark]*
- (ii) Construct the statements that will assign values of a 12,000 square metre field named “Back forty” to the `firstField` object. *[2 marks]*
- (iii) Construct the statements that will assign values, to the `firstField` object, indicating that it has a soil acidity value of 8 and the crop type is soybeans. *[2 marks]*

The program stores the data for all the fields in a single array of `FarmField` objects called `allFields[]`. The size of the array exceeds the number of `FarmField` objects. Unused elements at the end of the array are `null`. The current number of fields is stored in the variable `numberFields`.

- (b) Construct the method, `findLargest()`, that will return the size of the largest field in this array. It has been started below.

```
int findLargest(FarmField[] allFields, int numberFields)
{
    ...
}
```

[4 marks]

The `allFields[]` array is to be kept sorted so that `allFields[0]` is the `FarmField` object representing the largest field, `allFields[1]` the second largest, etc.

- (c) Construct the method, `insertField()`, that will insert a new `FarmField` object at the appropriate place in the `allFields[]` array. *[7 marks]*
- (d) Describe **one** advantage and **one** disadvantage of the size of the array `allFields[]` being greater than the number of `FarmField` objects. *[4 marks]*

This question requires the use of the Case Study.

3. (a) Outline **two** precautions that a mobile device user can take to prevent their data from being compromised if they lose their mobile device. *[4 marks]*
- (b) Outline **two** reasons why a phone designer might choose to integrate a solar power panel into a mobile device. *[4 marks]*
- (c) Identify **three** reasons why many Bluetooth devices can operate in the same office building without interfering with one another. *[3 marks]*
- (d) Compare the use of Ethernet with Bluetooth for data communication in smartphones. *[3 marks]*

Many mobile phones now have GPS integrated into them so that the user’s location can be sent to data services.

- (e) Discuss the advantages and disadvantages of letting a mobile device transmit the user’s location. *[6 marks]*
- (f) Smartphones contain several different kinds of memory.
- (i) State the type of memory in which applications (“apps”) would be stored when the mobile device is powered off. *[1 mark]*
- (ii) Describe the function of cache memory. *[3 marks]*
- (g) Discuss the effects that the widespread ownership of smartphones is having on students. *[6 marks]*
-