



Oxford Cambridge and RSA

AS Level Computer Science

H046/02 Algorithms and problem solving

Friday 8 June 2018 – Morning
Time allowed: 1 hour 15 minutes

**Do not use:**

- a calculator



First name										
Last name										
Centre number						Candidate number				

INSTRUCTIONS

- Use black ink.
- Complete the boxes above with your name, centre number and candidate number.
- Answer **all** the questions.
- Write your answer to each question in the space provided.
- Additional paper may be used if required but you must clearly show your candidate number, centre number and question number(s).
- Do **not** write in the barcodes.

INFORMATION

- The total mark for this paper is **70**.
- The marks for each question are shown in brackets [].
- Quality of extended responses will be assessed in questions marked with an asterisk (*).
- This document consists of **16** pages.



No calculator can
be used for this
paper

2

- 1 A user enters whole numbers into a computer program. Each number entered is placed onto a stack. The stack is created using an array with a maximum of 20 elements.

Part of the array, `numStack`, is shown when one number has been input.

<code>top</code>	1
------------------	---

index	stackItem
9	
8	
7	
6	
5	
4	
3	
2	
1	
0	20

The pointer, `top`, points to the next free space in the stack.

- (a) Complete the diagram below to show the state of `numStack` after the user inputs the following numbers in the order given:

22 13 2 59 1000

<code>top</code>	
------------------	--

index	stackItem
9	
8	
7	
6	
5	
4	
3	
2	
1	
0	20

[2]

(b) A function, `addItem`, takes a number as a parameter and adds the number to the stack. The function returns `true` if this was successful, and `false` if the stack is already full.

(i) Give **one** reason why a function is used instead of a procedure in this scenario.

.....
..... [1]

(ii) The parameter can be passed by value or by reference.

Describe what is meant by passing a parameter by value and by reference.

By value
.....
.....
.....

By reference
.....
.....
.....

[4]

(iii) The function `addItem` is written but is incomplete.

Complete the function, `addItem`.

```
function addItem (number)
    if top == ..... then
        return false
    else
        numStack[.....] = .....
        top = ..... + 1
        .....
    endif
endfunction
```

[5]

4

- (iv) The procedure, `calculate`, takes each item in turn from the stack. It alternately adds then subtracts the numbers until there are none left.

For example, if `numStack` contains:

2
6
5
12

It would perform $2 + 6 - 5 + 12$ and output 15.

```

01  procedure calculate()
02      total = 0
03      add = true
04      if top == 0 then
05          print("Stack empty")
06      else
07          total = numStack[top - 1]
08          top = top - 1
09          while top != 0
10              if add == true then
11                  total = total + numStack[top - 1]
12                  add = false
13              else
14                  total = total - numStack[top - 1]
15                  add = true
16              endif
17              top = top - 1
18          endwhile
19          print(total)
20      endif
21  endprocedure

```

5

Complete the trace table for the procedure `calculate`. The current array and pointer values when the procedure is called are on the first line of the trace table.

top	numStack						total	add	Output
	0	1	2	3	4	5			
5	20	2	6	12	8				

[6]

- 2 A games company has developed a game called Kidz Arrowz. The players throw an arrow at a target board and are awarded different points depending on which circle the arrow lands. Fig. 1 shows the board.

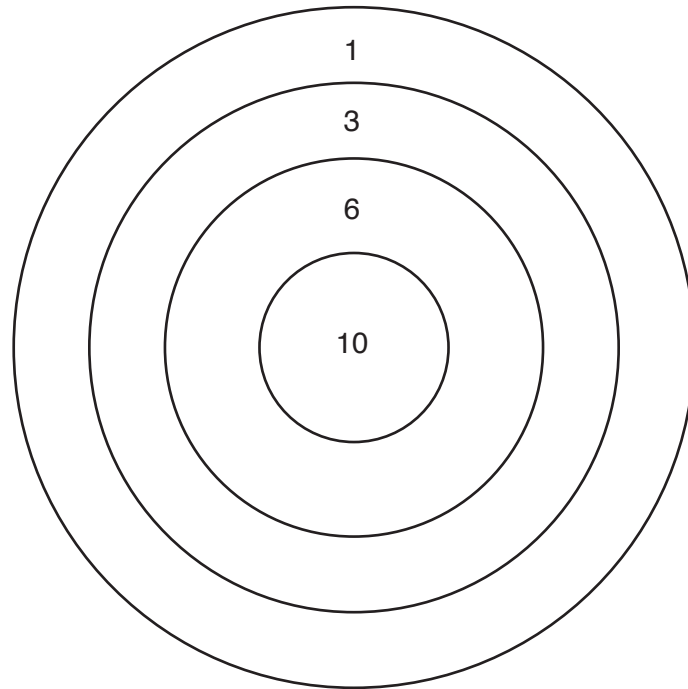


Fig. 1

A computer program is required to keep track of the scores for each competition. The user will enter the number of players, and the name of each player, in that competition to a maximum of 10. The program will then ask for the score of each player in turn. Each competition has 8 rounds, with each player throwing one arrow each round. The program will then display the total score of each player.

- (a) (i) The players are declared as a record structure:

```
record player(string playerName, integer totalScore)
```

Describe what is meant by a record structure.

.....

.....

.....

..... [2]

(ii) The records for the players are stored in a 1D array.

State why a 1D array is a suitable data structure for the records.

.....
..... [1]

(iii) Three data structures are arrays, records and stacks.

Identify **one** other data structure.

..... [1]

(b) The program is decomposed into multiple sub-programs, that each perform a specific task.

The array, `scores`, is declared as a global array of type record:

```
global array scores[10] of player
```

Explain why the array `scores` has been declared as global instead of local.

.....
.....
.....
.....
..... [2]

- (d) (i) The programmer has decided to use a bubble sort to sort the players' scores in descending order of score.

Describe the disadvantages of using a bubble sort.

.....

.....

.....

.....

.....

.....

..... [2]

- (ii) Despite the disadvantages, the programmer has decided to use a bubble sort for the players' scores.

Identify the characteristic of this problem that minimises the disadvantages of a bubble sort.

.....

..... [1]

- (iii) Write a procedure, `sortScores`, to perform a bubble sort on the global array `scores` to sort the players' scores into descending numeric order.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

[6]

(iv) An alternative sorting method is the insertion sort.

A procedure, `insertionSort`, has been written to sort an array `numbers`. The procedure is incomplete.

Complete the procedure.

```
procedure insertionSort()  
  for count = 0 to numbers.length - 1  
    position = .....  
    while position > 0 and numbers[position] < numbers[position-1]  
      temp = .....  
      numbers[position-1] = .....  
      numbers[position] = temp  
      position = .....  
    endwhile  
  next count  
endprocedure
```

[4]

(e) The programmer uses an Integrated Development Environment (IDE) to develop the program.

Describe how the IDE could be used to create the Kidz Arrowz program.

.....

.....

.....

.....

.....

.....

..... [3]

- 3 A country's national rail operator wants to represent their rail network on a computer system to keep track of the location of trains, and any problems on the network.

After studying the rail network, the operator uses abstraction to create the virtual representation.

- (a) (i) Define the term 'abstraction'.

.....
 [1]

- (ii) Identify **two** reasons why abstraction is needed in the rail network program.

1

 2
 [2]

- (iii) Describe **one** potential difference between the virtual and real rail network.

.....

 [2]

- (b) The rail operator provides an app for customers to purchase tickets. An array is used to store the names of the stations on the network. Customers must enter a departure station into the app.

The current contents of the array are shown:

Cavalry	Bridge	Walkway	Museum	Monument	Council House	Theatre	Cinema
---------	--------	---------	--------	----------	---------------	---------	--------

A linear search is used to check if the entered departure station exists in the array.

- (i) Identify **one** precondition that is needed before a binary search could be used with the station array.

.....
 [1]

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

END OF QUESTION PAPER

PLEASE DO NOT WRITE ON THIS PAGE

OCR

Oxford Cambridge and RSA

Copyright Information

OCR is committed to seeking permission to reproduce all third-party content that it uses in its assessment materials. OCR has attempted to identify and contact all copyright holders whose work is used in this paper. To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced in the OCR Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download from our public website (www.ocr.org.uk) after the live examination series.

If OCR has unwittingly failed to correctly acknowledge or clear any third-party content in this assessment material, OCR will be happy to correct its mistake at the earliest possible opportunity.

For queries or further information please contact the Copyright Team, First Floor, 9 Hills Road, Cambridge CB2 1GE.

OCR is part of the Cambridge Assessment Group; Cambridge Assessment is the brand name of University of Cambridge Local Examinations Syndicate (UCLES), which is itself a department of the University of Cambridge.