# 2.7 Programming Practice: Writing and Refining Programs: 30-minute Pack

OCR J277 · Calculators not allowed · about 30 minutes

Total Marks

Name: _____ Date: ____ / ____ / _____

**Answer ALL questions. Show all your working.**

**1** An array called scores stores the test scores of 8 students:
scores = [56, 78, 45, 90, 33, 67, 82, 71]

Write an algorithm that uses iteration to calculate and output the average score of the class.

**(Total for Question 1 is 5 marks)**

**2** Write an algorithm that repeatedly asks the user to input a whole number between 1 and 10 (inclusive), rejecting any input outside this range, until a valid number is entered. The algorithm should then output the valid number.

**(Total for Question 2 is 4 marks)**


© Revision Library | revisionlibrary.co.uk

**3** Write a function called applyDiscount that takes two parameters, price (a real number, in pounds) and percentageOff (an integer), and returns the price after the percentage discount has been subtracted.

**(Total for Question 3 is 4 marks)**


© Revision Library | revisionlibrary.co.uk

**4** Write a function called countVowels that takes a string parameter word and returns the number of vowels (a, e, i, o, u) in the string, ignoring case.

**(Total for Question 4 is 5 marks)**

---

**5** A function is intended to return the total of all the numbers in an array called numbers, but it returns an incorrect result:

```
FUNCTION calculateTotal(numbers)
    total = 0
    FOR i = 1 TO LEN(numbers)
        total = total + numbers[i]
    NEXT i
    RETURN total
ENDFUNCTION
```

**(a)** A function is intended to return the total of all the numbers in an array called numbers, but it returns an incorrect result: **(1)**


```
FUNCTION calculateTotal(numbers)
    total = 0
    FOR i = 1 TO LEN(numbers)
        total = total + numbers[i]
    NEXT i
    RETURN total
ENDFUNCTION
```


The array is 0-indexed, so a 5-element array has valid indices 0 to 4.

Identify the error in the function.

**(b)** Rewrite the FOR loop line so that the function works correctly. **(2)**

**(Total for Question 5 is 3 marks)**



© Revision Library | revisionlibrary.co.uk

**6** A shop stores its stock data in a text file called "stock.txt". The file lists each item on its own line, followed immediately by its quantity on the next line, repeated for every item, for example:

Kettle
14
Toaster
3

Write a program that:
- reads all the stock data from "stock.txt" into two parallel arrays, itemNames and quantities, keeping track of how many items were read in a variable itemCount
- includes a function called findItem that takes itemNames, itemCount and a search name as parameters, and returns the index of the item, or -1 if not found
- asks the user to input an item name to check
- uses findItem to look up the item and outputs its quantity, or a suitable message if the item is not found
- if the item is found and its quantity is less than 5, additionally outputs a low stock warning

**(Total for Question 6 is 8 marks)**


© Revision Library | revisionlibrary.co.uk

# Mark scheme · 2.7 Programming Practice: Writing and Refining Programs: 30-minute Pack

## Question 1

a total variable initialised to 0 before the loop **B1**

correct count-controlled loop that iterates over every element of the array (indices 0 to 7, or equivalent) **B1**

correct accumulation, total = total + scores[i], inside the loop **B1**

correct division of the total by 8 (the number of elements) after the loop, to calculate the average **B1**

OUTPUT of the average value **B1**

**Answer:**

```
scores = [56, 78, 45, 90, 33, 67, 82, 71]
total = 0
FOR i = 0 TO 7
    total = total + scores[i]
NEXT i
average = total / 8
OUTPUT average
```

## Question 2

correct condition-controlled loop structure with the exit-check after the input (e.g. REPEAT...UNTIL), or an equivalent entry-check loop that repeats correctly **B1**

INPUT statement placed inside the loop **B1**

correct UNTIL/loop condition testing number ≥ 1 AND number ≤ 10 (oe) **B1**

OUTPUT of the valid number placed after the loop has ended **B1**

**Answer:**

```
REPEAT
    INPUT number
UNTIL number >= 1 AND number <= 10
OUTPUT number
```

## Question 3

correct FUNCTION header with two parameters, e.g. FUNCTION applyDiscount(price, percentageOff) **B1**

correct calculation of the discount amount, e.g. discount = price * percentageOff / 100 (oe) **B1**

correct calculation of the new price by subtracting the discount from the original price (oe) **B1**

correct RETURN of the new price, with the function correctly closed using ENDFUNCTION (any correct logic; ignore minor syntax) **B1**

**Answer:**

```
FUNCTION applyDiscount(price, percentageOff)
    discount = price * percentageOff / 100
    newPrice = price - discount
    RETURN newPrice
ENDFUNCTION
```

## Question 4

correct FUNCTION header with one string parameter, e.g. FUNCTION countVowels(word) **B1**

a count variable initialised to 0 before iterating **B1**

correct iteration through every character of the string (e.g. FOR i = 0 TO LEN(word) - 1, oe) using a case-conversion function so case is ignored **B1**

correct selection testing the current character against the five vowels, combined with OR, and incrementing count when true **B1**

correct RETURN of count after the loop has ended **B1**

**Answer:**

```
FUNCTION countVowels(word)
    count = 0
    FOR i = 0 TO LEN(word) - 1
        letter = UPPER(SUBSTRING(i, 1, word))
        IF letter == "A" OR letter == "E" OR letter =
            count = count + 1
        ENDIF
    NEXT i
    RETURN count
ENDFUNCTION
```

## Question 5

(a) identifies that the loop runs from index 1 to LEN(numbers), which skips index 0 (the first element) and attempts to access index LEN(numbers), which is out of range (oe; accept identification of either half of the error) **B1**

**(a) Answer: The loop starts at index 1 instead of 0, so it misses the first element (index 0), and it runs up to LEN(numbers), which is one index beyond the last valid index (LEN(numbers) - 1), causing an out-of-range array access.**

(b) correct loop start changed to 0 **B1**

(b) correct loop end changed to LEN(numbers) - 1 **B1**

**(b) Answer: FOR i = 0 TO LEN(numbers) - 1**


© Revision Library | revisionlibrary.co.uk

## Question 6

| | |
|---|---|
| correct file opened for reading and a condition-controlled loop that reads until WHILE NOT EOF(stockFile) becomes false | **B1** |
| correct reading of a name line then a quantity line (cast to an integer with INT()) inside the loop, storing each into the respective array at index itemCount and incrementing itemCount (oe; accept declared fixed-size arrays with a running counter, or any other correct method of building the two arrays) | **B1** |
| correct findItem function header taking the array, itemCount and searchName as parameters | **B1** |
| correct iteration (e.g. FOR i = 0 TO itemCount - 1) and selection returning the matching index i inside the loop | **B1** |
| correct RETURN of -1 placed after the loop, only reached if no match was found | **B1** |
| correct INPUT for the search name, and a call to findItem storing the result in a variable | **B1** |
| correct selection: IF index == -1 THEN a not-found message is output, ELSE quantities[index] is output | **B1** |
| correct nested check, quantities[index] < 5, correctly nested inside the found branch, outputting a low stock warning when true | **B1** |

**Answer:**

```
itemCount = 0
DECLARE itemNames : ARRAY[0:49] OF STRING
DECLARE quantities : ARRAY[0:49] OF INTEGER
stockFile = OPENFILE("stock.txt", READ)
WHILE NOT EOF(stockFile)
    itemNames[itemCount] = READFILE(stockFile)
    quantities[itemCount] = INT(READFILE(stockFile))
    itemCount = itemCount + 1
ENDWHILE
```

**CLOSEFILE(stockFile)**

```
FUNCTION findItem(itemNames, itemCount, searchName)
    FOR i = 0 TO itemCount - 1
        IF itemNames[i] == searchName THEN
            RETURN i
        ENDIF
    NEXT i
    RETURN -1
ENDFUNCTION

INPUT searchName
index = findItem(itemNames, itemCount, searchName)
IF index == -1 THEN
    OUTPUT "Item not found"
ELSE
    OUTPUT quantities[index]
    IF quantities[index] < 5 THEN
        OUTPUT "Low stock warning"
    ENDIF
ENDIF
```


© Revision Library | revisionlibrary.co.uk