



MARCH 2016

PROGRAMMING

Instructions to candidates:

- a) Time allowed: Three hours (plus an extra ten minutes' reading time at the start – do not write anything during this time)
- b) Answer ALL parts of Question 1 (40 marks) and any FOUR other questions (15 marks each)
- c) Marks for subsections of questions are shown in []
- d) Spend about 1 hour on Question 1 and just under 30 minutes on each other question. Read CAREFULLY the particular instructions for Question 1
- e) State at the top of your answer FOR EACH QUESTION, where appropriate, the programming language and version you are using for that question. Different languages may be used
- f) Ensure that you pay particular attention to words underlined, in CAPITALS or in **bold**. FEW OR NO MARKS will be awarded to any question where these are ignored
- g) No computer equipment, books or notes may be used in this examination

General instructions for QUESTION 1: Answer all parts of this question.

- Provide high-level language solutions to each question in this section
- Do NOT provide any coding for anything outside the question – complete programs are NOT required
- Do not change the names of variables/file names in your answers
- Data only needs to be input into variables where the question specifically states 'INPUT'
- Any additional temporary variables YOU introduce must have clear meaningful names and be assigned with initial values

- 1. a) A useful calendar covering MANY years can be printed on a single page if only the first day of each month is displayed, e.g.

	FIRST OF THE MONTH											
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2016	Fri	Mon	Tue	Fri	Sun	Wed	Fri	Mon	Thu	Sat	Tue	Thu
2017												

So if it is known that April 1st 2016 is on a Friday, the other dates in April can be worked out fairly easily. Write part of a program to print a calendar for the years 2016 to 2036 with one line per year. The following array (M) is ALREADY held in memory showing the number of days in each month – February (*) will need adjusting upwards for leap years.

Month	1	2	3	4	5	6	7	8	9	10	11	12
Days	31	*28	31	30	31	30	31	31	30	31	30	31

HINT: Use a NUMERIC value for day. DAY = 0 for SUNDAY, DAY = 1 for MONDAY through to DAY = 6 for SATURDAY. Decide on your loops first. [20]

- b) Write a program that will prompt for input of a series of integers (stored in variable intNum). Count the integers as they are entered and store this value in variable intCount. Keep track of the accumulative total (intTotal).

The program ends when intTotal is greater than or equal to 100.

If the final input takes intTotal over 100, do not include this in the total and decrease the value of intCount.

Finally display the value of intCount. [8]

- c) Colours are symbolised by their initial letter. In this problem, only THREE colours are involved, RED (R), BLUE (B) and GREEN (G).

Input the symbols for TWO colours. Check that the colours are different before proceeding – re-input if necessary. Then, determine the third colour and output its code. So if 'R' and 'G' are input, 'B' will be output. [12]

2. Draw up an algorithm (pseudo-code or flowchart) to perform the following task:
 Input a decimal integer and convert it to binary using the standard manual method described below. This method shows conversion of decimal 22 but your algorithm must work for ANY decimal integer.
 Method – divide number repeatedly by 2 and store the remainder at each stage. The process stops when the number is reduced to zero. The remainders are then output in REVERSE order so you will need an array (or alternative structure) to hold each remainder at each stage.
- | | | |
|--------|-------------|-----------------------|
| 2) 22 | | |
| 2) 11 | remainder 0 | |
| 2) 5 | remainder 1 | |
| 2) 2 | remainder 1 | |
| 2) 1 | remainder 0 | |
| 0 | remainder 1 | so decimal 22 = 10110 |
- [15]
3. The following algorithm calculates discounts on customer accounts. Line numbers have been included to help you.
- a) Perform a dry run on this routine then answer the questions that follow:
- ```

10 Begin
20 T1 = 0
30 T2 = 0
40 Do while R <> "z"
50 Input N
60 Input R
70 Input A
80 If R = "y" then
90 D = A * 15/100
100 Else
110 D = A * 10/100
120 Endif
130 Print N, A, D
140 T1 = T1 + A
150 T2 = T2 + D
160 Loop
170 Print "Totals", T1, T2
180 End

```
- Data:  
 "Artworld ltd", "x", 1470, "Abc co.", "y", 300, "DesignCo", "y", 5000, "CopyPlus", "x", 550, "z", "z", 0[10]
- b) Which data item will cause the loop to exit? [1]  
 c) The discount for "x" accounts is to change to 12.5% and for "y" accounts 20%. What changes need to be made in the program? [2]  
 d) What effect, if any, would there be on the final output if "q" was mistakenly entered instead of "x" for the 4th input record? [2]
4. a) The following five points are factors to consider when writing a program in a **high-level language**.
- i The time to **develop** the logic and coding
  - ii The **number** of program statements in the source code
  - iii The time it takes to **compile** the program
  - iv The **run** time of the compiled program
  - v The size of **memory** to hold the compiled program
- Compare each of these factors to the **same program** being written in a low-level language. For EACH factor state whether the value would be higher, lower or the same for the low-level program. Justify EACH of your answers. [10]
- b) Outline the main stages of the compilation process. [5]
5. a) Discuss the differences between object oriented programming and procedural programming. [8]  
 b) Name a procedural programming language and an object oriented programming language and describe an application or program that each would be best suited for. [4]  
 c) Explain what the term **event-driven** means in programming, give an example. [3]
6. a) Explain why documentation is a necessary part of program development. [3]  
 b) Describe THREE different people that would need program documentation. For EACH person, explain what aspects of the documentation they would require and outline the reasons why they would need it. [12]

7. Each of the following is an error or an error message that might occur during development of a program. For EACH:
- state whether it occurs at RUN-TIME or during COMPILATION
  - explain HOW the error can be corrected so that it does not occur when an IDENTICAL run is performed immediately afterwards
- a) DIVISION BY ZERO
  - b) SUBSCRIPT ERROR
  - c) Incorrect final totals on an invoice
  - d) FILE NOT FOUND
  - e) PRINT is used as a variable name
- [3 each]
8. Outline THREE **different** program design tools. For EACH, describe their purpose, discuss the advantages and illustrate an example.  
Note – the advantages should be **different** for each method described.
- [5 each]