BCS THE CHARTERED INSTITUTE FOR IT

BCS HIGHER EDUCATION QUALIFICATIONS
BCS Level 4 Certificate in IT

SOFTWARE DEVELOPMENT

Wednesday 19th April 2023 – Afternoon

Time: TWO hours

Section A and Section B each carry 50% of the marks.
You are advised to spend about 1 hour on Section A (30 minutes per question)
and 1 hour on Section B (12 minutes per question).

Answer any Section A questions you attempt in Answer Book A
Answer any Section B questions you attempt in Answer Book B

The marks given in brackets are **indicative** of the weight given to each part of the question.

Calculators are **NOT** allowed in this examination.

Section A
Answer 2 questions (out of 4). Each question carries 30 marks.

A1.

Consider the following algorithm for converting a decimal number into a binary number:

- Obtain a decimal number from the user and store it in an integer variable named N.
- Perform an integer division to divide this number by 2 and store the result in N.
- Store the remainder in the next available position in an array.
- Repeat the above two steps while N is greater than zero.

The result is obtained by printing the array in reverse order.

a) Trace the steps in this algorithm to show what happens step-by-step when the initial decimal number is 10.

(10 marks)

b) Provide code in a language of your choice that shows how this algorithm might be implemented as a function or procedure. Your answer should include the code necessary to invoke the function or procedure. This will prompt the user for a decimal value and output the answer as 8 binary digits. You can assume that the user will only enter values of N in the range 0 <= N <= 255.

(10 marks)

c) With respect to this example, explain the difference between calling a function by value and by reference.

(5 marks)

d) Explain the difference between functions and procedures.

(5 marks)

A2.

a) Describe the key principles and controls used by interface designers in **each** of the following areas:

i) Navigation design; **(6 marks)**
ii) Input design; **(6 marks)**
iii) Output design. **(6 marks)**

b) Outline the key techniques that a software developer can employ to evaluate the usability of a User Interface.

(12 marks)

B9.

Compare and contrast **each** of the following pairs of related terms:

i) Tuple vs List;
ii) Operating System vs Application Software;
iii) Command Line and Graphical User Interface.

(12 marks)

B10.

a) Give a definition of **each** of the following program control structures: *sequence*, *selection* and *iteration*.

(3 marks)

b) Using example code (in any language with which you are familiar) describe **THREE** different types of iteration/looping statements.

(6 marks)

c) Explain using an example what is meant by an infinite loop in a computer program.

(3 marks)

B11.

a) What is acceptance testing and what are the expected outcomes of acceptance testing from a software developer perspective?

(6 marks)

b) Explain the concept of software maintenance and briefly outline the role of software maintenance during the lifetime of a software product.

(6 marks)

B12.

a) Describe the characteristics of a CSV file.

(4 marks)

b) A given CSV file contains three values in each line in the file. Use pseudocode to design a program which searches sequentially through the file reading one row/record at a time for a particular value in the first field. If the value matches the search value, it adds 1 to a running total. The running total is output when all rows/records have been read and searched.

(8 marks)

END OF EXAMINATION

B6.

Consider the following lists of data. You can assume these represent a small sample of larger data sets:

List1: 3 16 13 9 15 12 2 10 14 8 7 1 6 4 11 ... …
List2: 14 1 5 6 13 12 3 11 15 4 7 8 10 2 16 … …

Also consider the following processing tasks (Task1 and Task2) on the above lists:

Task1: Find data elements from list 1 that are also present on list 2.
Task2: Merge the two lists to remove duplicate values.

a) Outline in no more than **TWO** sentences why it is beneficial to sort the lists of data before carrying out Task1 above.

(2 marks)

b) Describe using pseudocode how Task1 is processed using sorted lists of data.

(6 marks)

c) Describe the Big O notation and briefly explain how, in this case, it indicates that using sorted data would increase the speed of processing of Task2.

(4 marks)

B7.

Write short notes on the following software utilities and explain the association (if any) they have with each other during software execution:

i) Linker; **(3 marks)**
ii) Loader; **(3 marks)**
iii) Compiler; **(3 marks)**
iv) Interpreter. **(3 marks)**

B8.

You have been asked to debug a computer program, that was written by someone other than yourself, that compiles and runs but does not work as expected.

a) Write out the steps you would take to debug the program if you only had the following documentation available:

- Print outs of the Source code;
- Screen shots showing the program execution;
- A flowchart outlining the program design;
- Results of running unit tests on different parts of the code.

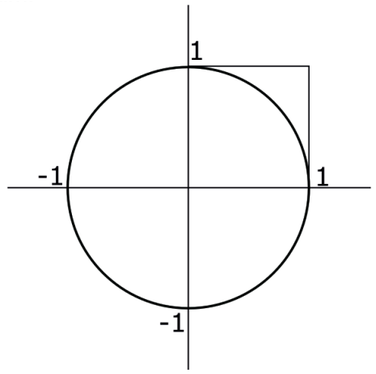
(6 marks)

b) Explain the role of black box testing and give a couple of examples of documentation arising from black box testing that could be used to help debug the program.

(6 marks)

A3.

Consider the following diagram:



The circle shown has a radius of 1. The area of the circle can be calculated using the formula πr^2 and is therefore equal to π.

Consider the top right-hand quadrant of the diagram. A quarter of the circle is in this quadrant with area $.25\pi$. The square in this quadrant has a side which is one unit in length. Therefore, the area of the square is 1.

Consider a point x, y where 0 <= x <= 1 and 0 <= y <= 1.

The distance from the centre of the circle to the point x, y is given by: $\sqrt[2]{x^2 + y^2}$

If this distance is less or equal to 1 the point x, y is within the circle, if it is greater than 1 it is outside the circle. Given the possible values for x and y, every point x, y will lie within the square.

If a set of points are created by selecting random values of x and y, some of these points will lie within the circle quadrant whilst the others lie within the square but outside the circle quadrant. If the values are truly random then, as more points are generated, the number of points within the circle quadrant divided by the total number of points generated will approach the value calculated by dividing the area of the quadrant by the area of the square. As already stated, this number is $.25\pi/1 = .25\pi$.

a) Given a function random() which returns a random number n, such that 0 <= n <= 1, write a program which calculates an approximate value of π.

(20 marks)

b) Explain how you might test this code and state whether a white box or a black box testing methodology would be appropriate.

(5 marks)

c) What factors might influence the accuracy of the output from the program?

(5 marks)

[Turn Over]

A4.

a) What techniques should a programmer use to make their code as readable as possible?

(10 marks)

b) Explain the difference between user documentation and system documentation.

(10 marks)

c) Discuss the design issues you would consider when constructing a web site to document a software system.

(10 marks)

Section B
Answer 5 questions (out of 8). Each question carries 12 marks.

B5.

A geometric progression is a sequence of terms in which each term is a constant multiple of the preceding term.

2, 6, 18, 54, 162 is an example of a geometric progression.

This can be expressed algebraically as follows:

ar^0, ar^1, ar^2, ……., ar^{n-1}

Where “a” is the initial value, “r” is the common ratio and “n” is the number of terms. In the example above a = 2, r = 3 and n = 5.

a) State the values of a, r and n for the following geometric progression:

64, 32, 16, 8, 4, 2, 1, 0.5

(2 marks)

b) Write a function called fnSumofGP(a ,r, n) that returns the SUM of the terms that form the geometric progression, where a, r and n are the required input parameters of the function. For example, using geometric progression in part a) above, the SUM returned is: 127.50

(5 marks)

c) Write a program that checks if n >=1000 and r is between -1 and +1. If TRUE, the SUM of a geometric progression is computed using the following expression.

SUM = a / (1-r)

If FALSE, the program calls the function fnSumOfGP as described in part b).

(5 marks)

[Turn Over]