BCS THE CHARTERED INSTITUTE FOR IT

BCS HIGHER EDUCATION QUALIFICATIONS
BCS Level 4 Certificate in IT

SOFTWARE DEVELOPMENT

Wednesday 18th March 2020 - 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 the Section A questions you attempt in Answer Book A
Answer the 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) in Answer Book A. Each question carries 30 marks.

A1.

A company sells four products in five regions. The sales of each product in each region have been reported as follows:

	Product 1	**Product 2**	**Product 3**	**Product 4**
Region				
1	189	28	132	12
2	123	89	356	18
3	356	54	128	28
4	104	13	108	35
5	205	18	102	23

Write a program in a language of your choice to do the following:

a) Store the above values in a suitable data structure.

(5 marks)

b) Calculate the total number of sales of each product and output these as a percentage of the total number of sales.

(12 marks)

c) Output the product number (e.g. Product 2) of any product whose sales exceed 50% of the total sales. If there are no products which fulfil this condition output a message which indicates this.

(5 marks)

d) Output the product number and region number combination which has the highest number of sales. (Note: There may be more than one region and product combination with the same number of sales).

(8 marks)

For each of the above, marks will be given for the clarity of your code; you should use comments and meaningful variable names to help with this.

[Turn Over]

A2.

a) Consider the code below and write it out in a clearer human readable form.

```
counter = 0;Get num;for(i=2;i<=num/2;i++){if(num %
i==0){counter = counter+1;}}if(counter==0 && num!= 1)
Write(num + " is a prime number"); else Write(num + " is not a
prime number");
```

(10 marks)

b) Referring to the code in part a) above find and write out the following and explain the role they play in making the program work:

i) All the identifiers;
ii) All the operators;
iii) A conditional (logical, boolean) expression;
iv) The iterative (repetitive, loop) statement.

(12 marks)

c) Write out the code from a) replacing the for-loop with an equivalent while-loop.

(8 marks)

[Turn Over]

A3.

a) Trace the behaviour of the following code when the value provided for n is 7.

```
int n, i, j convertedNo=0, dn;
write("Enter a number to convert: ");
get n;
dn = n;
i = 1;
for(j = n; j > 0; j = j / 2){
  convertedNo = convertedNo + (n % 2) * i;
  i = i * 10;
  n = n / 2;
}
write("The converted version of ", dn, " is ",
convertedNo);
```

(15 marks)

b) Describe in a single sentence the purpose of the code in part a).

(5 marks)

c) The following algorithm produces the same result as the code given in part a) using an array to hold each digit of the result. It then outputs the array.

1. Read the number to be converted n
2. Store the remainder when n is divided by 2 into an array
3. Set n = n/2
4. Repeat steps 2 and 3 until n = 0
5. Output the array in reverse order

Write code in a language of your choice to implement this algorithm. Assume that 0 <= n <= 256.

(10 marks)

A4.
For each of the following parts, write code in a language of your choice. You may make use of a pre-written function *len* which takes an array of integers as its parameter and returns the number of integers in the array.

a) Write a function *copyArray* which takes an array of integers as its parameter and returns a copy of the array.

(6 marks)

b) Write a function *sumArray* which takes an array of integers as its parameter and returns the sum of all the integers in the array.

(6 marks)

c) Write a function *maxMin* which takes an array of integers as its parameter and returns a second array containing the maximum and minimum values in the first array.

(6 marks)

d) Write a function *duplicates* which takes an array of integers as its parameter and returns the number of elements that appear twice (and no more than twice) in the array. You may make use of a pre-written function *sort* which takes an array of integers as its parameter and returns an array which contains the elements of the array passed to the function sorted in ascending order.

(12 marks)

[Turn Over]

SECTION B

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

B5.

The following algorithm describes the conversion of a decimal number (base 10) to an equivalent number with a different base, for example binary, octal (base 8) using modulus division.

Step 1: Modulus divide the decimal number by the designated base producing a quotient and a remainder.
(For example, 5 "modulus divide" by 2 returns 1 as the remainder and 2 as the quotient.)
Step 2: The remainder is then stored in memory as a single character.
Step 3: Repeat step 1 again until no further divisions are possible.
Step 4: The converted number is output as the reverse of the string of stored characters from step 2.

a) Step through the above algorithm showing how 69 in decimal is converted to 105 in octal.

(4 marks)

b) Explain what is meant by the term '**iteration**'. Explain why iteration would be necessary in the coding of the above algorithm.

(4 marks)

c) What is the most efficient data structure to store the remainder digits (Step 2) so that they are output in the correct order (Step 4)? Justify your answer.

(4 marks)

B6.

a) Rewrite the following sequence of code to expose the structure of the nested IF statements and thus make it easier to read.

```
if age > 11 then if age < 18 then print "junior" else
print "adult" endif else print "child" endif
```

(4 marks)

b) Express in English the **THREE** conditions that the above code computes.

(3 marks)

c) Rewrite the code in part a) by replacing the nested IF statements with Boolean AND/OR and ELSEIF statements resulting in more readable code.

(5 marks)

[Turn Over]

B7.

a) Compare and contrast user documentation with technical documentation for a software product/application in terms of who uses it, its content, and its structure and how it is delivered and presented.

(7 marks)

b) Describe the purpose of using naming conventions in program code. Give an example of how naming conventions are used in program code.

(5 marks)

B8.

For a software developer, establishing the “user requirements” is a key stage in the software development life cycle.

a) Briefly explain what is contained in a “user requirements specification”?

(4 marks)

b) List a range of techniques that are used to collect/elicit user requirements. Comment on the benefits and any drawbacks in using these techniques.

(8 marks)

B9.

a) Using a couple of sentences and an example describe the function of **EACH** of the following types of User Interface:

- Dashboard;
- Form;
- Command Line Interface (CLI).

(6 marks)

b) Discuss the advantages and disadvantages of using a CLI as an alternative to using a dashboard or a form.

(6 marks)

B10.

Developing high quality software depends on meeting professional software standards and criteria.

a) State **FOUR** quality criteria that are used and outline how they contribute to the development of high-quality software.

(8 marks)

b) A company called XYZ has discovered that a former employee has been using highly valuable software that was developed by other employees of XYZ.

Discuss what legal and/or ethical issues are relevant in the above scenario.

(4 marks)

[Turn Over]

B11.
A text file contains records which consist of three values in three fields. A program is required to search through the file reading one row/record at a time for a particular value in the first field. If the field matches the search value add 1 to a running total. The running total is output until all rows/records have been read and searched.

Given the above requirement:

a) Describe the type of search algorithm that would be used to solve this problem.

(4 marks)

b) Discuss what type of file organisation would be most efficient for supporting the search process.

(4 marks)

c) Why might a csv (comma separated value) data structure be used to store the data in this file?

(4 marks)

B12.

a) Explain the differences between a run time error and a logical error in program code.

(4 marks)

b) Give an example of a **run time error**.

(2 marks)

c) Give an example of a **logical error**.

(2 marks)

d) A **debugger** is a software tool that can find and fix errors in code. Explain how a programmer uses a debugger to find and fix errors in their code.

(4 marks)

END OF EXAMINATION