**Wednesday 08 February 2023**
**14:00-16:30 GMT**
**Duration: 2 hours**
**Additional time: 30 minutes**
**Timed exam – fixed start time**

**BSc Software Engineering (Graduate Apprenticeship)**

# PRACTICAL ALGORITHMS: COMPSCI1021

**Answer All Questions**

**This examination paper is an open book, online assessment and is worth a total of 100 marks.**

# Question 1 (30 Marks)

## *Algorithms, Searching & Sorting, and Complexity*

1a. Consider the following algorithm described in pseudo-code. It has two inputs: an array **array** of positive integers and one integer **output_size**. It locally uses a Stack data structure called **stack**. It creates and returns a list called **output**.

```
define some_algo (array, output_size)
    output = []
    stack = new Stack

    if length of array is less than output_size then
       zeros_to_pad = output_size–length(array)
       append zeros_to_pad zeros to array

    for i from 0 to output_size-1
       if array[i] is negative then
             stack.push(0)
       else
             stack.push(array[i])

    for i from 0 to output_size -1
      output.append(stack.pop())

    return output
```

Answer each of the following concerning this algorithm:

(i) Describe briefly what this algorithm does. That is, given a certain input, how does this algorithm transform it? (Note that you are not asked to describe line by line what the code does. Try to describe what this algorithm does as a whole.)

(ii) What is returned value when this function is called as follows: `some_algo([1, 2, 3, 4, 0, 5, 6], 9)`

(iii) Using the big-O notation, characterize the running time of the above algorithm in terms of **n**, the number of integers in **array**. Show your working.

[4+4+6]

1b. Consider a 6-element array being sorted into ascending order using the merge sort algorithm. This is the state of the array before the *final* **merge** is applied:

`A = [4, 8, 12, 1, 2, 19]`

Show a step-by-step application of the final `merge` operation on this array. That is, show what happens when `merge(A, 2, 5)` is applied. You should show the state of the entire A array, as well as any temporary arrays created for the `merge` operation, for each step. Also, write a short comment to indicate what has happened in each step.

[8]

1c. Using Merge-sort as an example and with reference to its pseudo-code given below, explain what is meant by *recursive functions*. Show that the worst case running time of Merge-sort is O(n $\log_2 n$).

```
merge_sort(array, start, end)
  if start < end
    mid := (start+end)/2
    merge_sort(array, start, mid)
    merge_sort(array, mid+1, end)
    merge(array, start, mid, end)
```

[8]

# Question 2 (25 Marks)

## *Data Collections*

2a. Given A = {}, B = {m, n, o, p} and C = {p, q}, P(X) denotes the power set of X, and |X| denotes the cardinality of set X, determine:

(i) $P(A \cup C)$
(ii) $|P(A \cup B \cup C)|$
(iii) $A \times B$
(iv) $A'$

[6]

2b. In the context of set theory:
i) Explain in your own words what is the difference between a *set* and a *sequence*.
ii) Give one example of a data collection that is organized as a set, and another example where a data collection is organized as a sequence. The chosen examples should highlight the key difference(s) between sets and sequences.

[3+3]

2c. Suppose you wish to implement an *ordered list* such that *searching* the list is efficient. Given the choice of data structure to implement is between an *array* or *doubly linked list*, which one would you choose, and why? Your answer should incorporate a discussion of the big-O complexity of both approaches.

[6]

2d. Suppose we are hashing integers into a 9-bucket hash table using the hash function:

```
int hash(int i){return i % 9;}
```

The call `put(i)` will put the integer `i` into the hash table using the above hash function.
Show what steps will be performed, and the resulting open hash table, using *linear probing*, if the sequence of calls `put(2), put(11), put(4), put(83), put(125), put(216)` are made on an initially empty hash table.

[7]

# Question 3 (15 Marks)

## *Sequences, Induction, and Recurrence*

3a. Let $\{a_n\}$ be a sequence that satisfies the recurrence relation $a_n = a_{n-1} + 2a_{n-2}$ for n = 1,2,3,4,… . Suppose that $a_0 = 1$ and $a_1 = 1$

What are the terms $a_0$, $a_1$, $a_2$, $a_3$ and $a_4$ and $a_5$ the sequence $\{a_n\}$?

[4]

3b. Let $\{b_n\}$ be a sequence that satisfies the recurrence relation $b_n = 2b_{n-1}$. Suppose $b_0 = 1$. Find its closed formula using the iterative method.

[5]

3c. Using mathematical induction, prove that for any positive integer n, $n^3 + 2n$ is divisible by 3.
*[ Hint: $(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$ ]*

[6]

# Question 4 (15 Marks)

## *Counting and Probability*

4a. Suppose a country has a post code format Z-XXX-NNN, where:

Z represents the capital English alphabet 'Z' itself.
X represents any capital letter of the English alphabet, except these three: 'Z', 'O' and 'I'
N represent a digit from 0 to 9

As an example, the following are valid post codes under this scheme: Z-BFR-453, Z-QQE-897, Z-ABW-867

How many distinct postcodes are possible under this scheme?

[4]

4b. High performance computing often requires the use of a cluster of computers. Suppose you are asked to create such a cluster that consists of 10 computers. You can choose from the two types of computers, which are available in different quantities as shown below:

Type A computer (CPU-only based): 5 available computers
Type B computer (CPU+GPU based): 9 available computers

How many ways are there to create a cluster of 10 computers, such that there is an equal number of Type A and Type B computers.

[6]

4c. What is the conditional probability that a randomly generated decimal number of length four contains *at least* two consecutive 0s, given that the first digit is 9? (E.g. 9100, 9002, etc). Explain your answer.

[5]

# Question 5 (15 Marks)

## *Formal Reasoning*

5a. Using either truth tables or laws of logical equivalences, show that the following equivalence is valid:

$$\neg(p \wedge q) \vee \neg q \equiv \neg(p \wedge q)$$

[6]

5b. Suppose we have the following predicates:

- $L(x)$: x is an email from a legitimate address
- $S(x)$: x is spam

Express this English statement in logical formulae using the predicates given above: “No email from a legitimate address is spam.”

[4]

5c. Using the definition of odd and even integers*, prove the following statement is true for all integers n: “if n is odd, then $n^3$ is odd”. Use either direct proof, or proof by contradiction.

[5]

**Hints:*
*if n is even, then n = 2k for some integer k.*
*if n is odd, then n = 2k+1 for some integer k.*
*$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$*