NATIONAL JUNIOR COLLEGE
Mathematics Department

General Certificate of Education Advanced Level
Higher 2

COMPUTING 9569/01

Paper 1 Written

29 Sept 2022

2 hours

Additional Materials: Pre-printed A4 Answer Booklet

READ THESE INSTRUCTIONS FIRST

An answer booklet will be provided with the question paper. You should follow the instructions on the front cover of the answer booklet. If you need additional answer paper ask the invigilator for a continuation booklet.

There are 5 questions totalling 65 marks.

Answer **all** questions.
Approved calculators are allowed.

The number of marks is given in the brackets [ ] at the end of each question or part question.

This document consists of 8 printed pages and 0 blank pages.

1 The following Python code defines a recursive function that performs a specific task.

```
1  LETTERS = ["0","1"] //a list of characters
2
3  FUNCTION DoIt(n: INTEGER) RETURNS List
4  // List is a dynamically sized array
5    IF n = 1 THEN
6      RETURN LETTERS
7    ELIF n < 1 THEN
8      RETURN [] // returns empty List
9    ENDIF
10   ret = [] // creates an empty List
11   result = DoIt(n-1)
12   FOR letter in LETTERS DO //iterate over each element
13     FOR s IN result DO
14       Append( ret, Concatenate(letter, s) )
15 //Concatenates letter with s and append the resultant string
16 //into the List ret
17     ENDFOR
18   ENDFOR
19   RETURN ret
20 ENDFUNCTION
```

(a) There are two components in a recursive function. Identify, using line numbers, the two components of a recursive function in the code shown above. [2]

(b) What task does the function `DoIt()` perform ? [1]

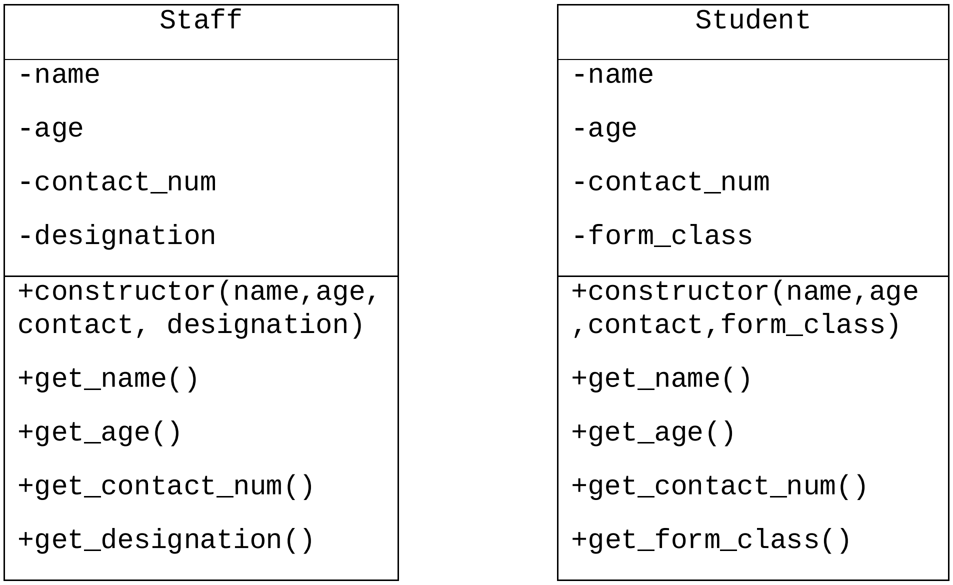
(d) Trace the execution of the code when `DoIt(3)` is executed by completing the trace table shown below:

Call Frame	**result**	**ret**	**RETURN**
DoIt(3)			
DoIt(2)	["0","1"]		
DoIt(1)			["0","1"]

[3]

(e) What is the run time complexity of the function `DoIt()`? Explain intuitively how you arrive at your answer. [2]

2 The Staff and Student Abstract Data Types (ADTs) are used to model the behaviour of staff and students in a school. The UML class diagram show below is being drawn to represent the two ADTs:



```
Staff
-name
-age
-contact_num
-designation
+constructor(name,age,
contact, designation)
+get_name()
+get_age()
+get_contact_num()
+get_designation()
```

```
Student
-name
-age
-contact_num
-form_class
+constructor(name,age
,contact,form_class)
+get_name()
+get_age()
+get_contact_num()
+get_form_class()
```

- Indicates a private attribute/method

+ Indicates a public attribute/method

(a) Identify and explain the problems or inefficiencies that you may encounter if the above design is implemented in a school management system. You should consider the practical aspects as well as the Object Oriented approach in the design. [4]

(b) Re-draw the UML diagram to reflect on the improvement that you can made based on the problems and inefficiencies that you have identified in part (a). [2]

A dynamic data structure, List, can be used to store a list of objects of different types. For example a list of Staff and Student objects can be stored in a List, `ALL,` as follows:

```
ALL ← [
    Staff("Mr Sam",34,"234","Teacher"),
    Student("Joe",15,"2345","3A"),
    Staff("Ms Lam",24,"234","Teacher"),
    Student("Lynn",16,"2345","4D"),
    Student("Zuk",17,"2345","4D"),
    Student("Jill",15,"2345","3A"),
    Staff("Mr Bala",30,"234","Admin")]
```

(c) Using pseudocode, describe a bubble sort algorithm that will re-order the List as follows:

- All the Staff objects should appear before the Student objects.
- The Student objects should be ordered by ascending order of their age.

The re-ordered List, `ALL`, should look like this:

```
[
    Staff("Mr Sam",34,"234","Teacher"),
    Staff("Ms Lam",24,"234","Teacher"),
    Staff("Mr Bala",30,"234","Admin"),
    Student("Joe",15,"2345","3A"),
    Student("Jill",15,"2345","3A
    Student("Lynn",16,"2345","4D"),
    Student("Zuk",17,"2345","4D")
]
```

[7]

The insertion sort algorithm can also be used to perform the sorting described in part (c).

(d) Identify two similar characteristics/features of the bubble and insertion sort algorithm. [2]

(e) Describe a scenario whereby the insertion sort will have a better run time efficiency than the bubble sort algorithm. [2]

(f) Given two sorted list consisting of `Student` and `Staff` objects, ordered according to the requirements described in (c) above.

Write an algorithm, using pseudocode, to merge the two list into a single sorted list based on the order described in (c) above. [3]

There is a requirement to search a sorted list of `Student` and `Staff` objects by their name and output the contact number if the name is found in the list. The list is ordered according to the requirements described in (c) above. There are two proposed search algorithm, linear and binary search.

(g) Which search algorithm do you propose to use ? Explain why you choose that algorithm over the other, and the run time efficiency of the chosen algorithm. You do not need to describe the algorithm. [3]

3

```
1  def insert_sort(A: list):
2      for i in range(1, len(A)):
3          j = i
4          item = A[i]
5          while j > 0 and A[j-1] > A[j]:
6              A[j] = A[j-1]
7              j-=1
8          A[j] = item
```

(a) The Python code above uses the insertion sort algorithm to sort a Python List in ascending order. What is the average run time complexity of the algorithm used to perform the task? [1]

(b) The Python function, `insert_sort()`, above does not execute correctly. Identify and correct the logical error in the code above by identifying the line number and the logical error. [2]

```
1   def quick_sort(A: list) -> list:
2       if len(A) < 2:
3           return A
4       else:
5           pivot = A[0]
6           left=[]
7           right=[]
8           for item in A[1:]:
9               if item < pivot:
10                  left.append(item)
11              else:
12                  right.append(item)
13          return q_sort(left) + [pivot] + q_sort(right)
```

The code above shows a correct implementation of the quick sort algorithm.

(c) What is the average run time complexity of the quick sort algorithm ? [1]

(d) Suggest an improvement on the code implemented for the quick sort algorithm so that a worst-case input can be handled more efficiently. [2]

On the same computer with the same input List,

```
 [3, 2, 2, 7, 9, 2, 1, 6, 6, 2],
```

the execution time for the quick sort code is determined to be 10.5 microsecond (µs), while the execution time of the insertion sort code (after the error is fixed) is 5.5 microsecond (µs). Both measurements are perform using the Python `timeit` module.

(e) Explain and justify the run time complexities of the two algorithms shown above with the measured execution time of the code being implemented. [3]

4 The Singapore Wine Appreciation Club (SWAC) is a member only club which has a collection of wines sourced from the finest wineries all over the world. The objectives of the club are to educate its members in the fine art of wine tasting and to provide a knowledge base of all the wines that has been produced in the world.

SWAC wishes to build an application to manage the information about its members as well as to record the information they gather about various wines in a relational database. The club is big enough so that there are several members with the same name. A membership card with a unique number is issued to identify each member. The email and contact addresses of each member is also recorded.

At most once a week, SWAC organises a wine tasting session. At each session, the attending members taste several bottles of wines. Each member records for each bottle his or her evaluation of the quality (a discrete score between 1 and 5, with 1 being very bad to 5 being very good) of each wine that he or she tastes. The evaluation may differ for the same wine from one member to another. Actual quality and therefore evaluation also vary from one bottle to another bottle of a given wine. Every bottle that is opened during the tasting session is finished during that session. These evaluations need to be recorded in the database and each member is only allow to submit one evaluation for each bottle of wine that he/she has tasted.

Each wine is identified by its name (eg, "Parade D'Amour") and vintage (eg, 1990). Other information of interest about the wine is the degree of alcohol (eg, 11%), and the country it comes from ("eg, France"). The same wine produced in different year (its vintage) may have different characteristics.

Generally, there are or have been several bottles of the same wine kept in the cellar of SWAC. For each wine, the bottles are numbered. For example, the cellar has twenty bottles of "Parade D'Amour" from 1990 numbered 1 to 20. For educational purposes, SWAC may also want to record wines for which it does not own bottles.

The bottles are either available in the cellar, or they have been tasted and drunk during the tasting sessions.

(a) Draw an entity-relationship diagram (ERD) that describes the main entities and their relationships based on the description given above. The data model should be in third normal form (3NF). [5]

(b) A table description can be expressed as:

`TableName (Attribute1, Attribute2, Attribute3, …)`

The primary key is indicated by underlying one or more attributes. Foreign keys are indicated by using a dashed underline.

Write the table descriptions for the required tables in the database so that they are in third normal form (3NF) and meet all the constraints and requirements for the database application described above. [5]

(c) Write the SQL statement to list all the different countries that the wines come from. [2]

(d) Write the SQL statement to print the names of members who have made more than one wine evaluation. [4]

(e) Write the SQL statement to delete the records of bottles that have been consumed during the wine tasting sessions. [1]

(f) When executing the SQL statement you have written in (e) on the database, an error occurs. Describe and explain the cause of the error. [2]

(g) The database application and the database is implemented and run on a single server machine. The database application must be running continuously at all time and cannot be shutdown. The storage capacity on the server is running low. What measures do you propose to ensure that the database application can continue to operate in the future. [2]

5 A sparse matrix is a matrix in which more than half of the elements in the matrix are zero. For example, the following is a 4 × 4 sparse matrix with 4 non-zero values and 12 zero values:

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 5 & 8 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 6 & 0 & 0 \end{pmatrix}$$

Using two-dimensional arrays to represent very large sparse matrices waste a lot of memory space. In the example above, we need to use 16 memory slots to store 4 values.

Describe a way to represent a sparse matrix so that we can reduce the amount of memory space used. [4]

END OF PAPER