|  | NATIONAL JUNIOR COLLEGE  Mathematics Department  General Certificate of Education Advanced Level  Higher 2 | | | |
| --- | --- | --- | --- | --- |
| **COMPUTING**  Paper 1 Written | | | | **9569/01**  **2 October 2020**  **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 **6** questions totalling **70** 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 5 printed pages and 3 blank pages.  NJC Mathematics 2020 **[Turn over** | | | | |

| **1** | The following Python code defines a recursive function that performs a specific task.   | 1  2  3  4  5  6  7  8 | def recur(s):  if len(s) == 1:  return s[0]  ret=[]  for i in range(len(s)):  for ss in recur(s[:i]+s[i+1:]):  ret.append(s[i] + ss)  return ret | | --- | --- | | | |
| --- | --- | --- | --- | --- | --- |
|  | **(a)** | Explain what is a recursive function ? | [2] |
|  | **(b)** | Explain the difference between an iterative solution and a recursive solution.to a problem. | [2] |
|  | **(c)** | Trace the recursive calls when the following code is executed:   | recur("ABC") | | --- |   You should clearly indicate the value of the argument used in each recursive call and the value returned, including the final result. | [4] |
|  | **(d)** | What task does the function recur perform ? | [1] |
|  | **(e)** | What is the run time complexity of the function recur ? Explain intuitively how you arrive at your answer. | [2] |

| **2** | **(a)** | Use pseudocode to describe a recursive function that performs the mathematical exponential function:  , *where x is a rational number and n is an non-negative integer*.  You are allow to use **only** the four basic arithmetic operators: and .  Your algorithm should have the most efficient time complexity. | [5] |
| --- | --- | --- | --- |
|  | **(b)** | State the time complexity of the algorithm and explain why your algorithm is the most efficient. | [2] |

| **3** | **(a)** | Using an insertion sort, show how the following list of strings can be sorted in alphabetical order:  "MYS", "AUS", "CAN", "SGN", "JPN". | [3] |
| --- | --- | --- | --- |
|  | **(b)** | Below is a bubble sort pseudocode for sorting a Python List in ascending order.  Note that declaration statements are omitted in the pseudocode.  01 NoSwaps 🡨 False  02 WHILE NoSwaps = FALSE  03 NoSwaps 🡨 TRUE  04 UpperBound 🡨 ListLength - 1  05 FOR Posn 🡨 0 TO ... **A** ...  06 IF List[Posn] > ... **B** ...  07 THEN  08 // Swap  09 NoSwaps 🡨 ... **C** ...  10 Temp 🡨 List[Posn]  11 List[Posn] 🡨 List[Posn + 1]  12 List[Posn + 1] 🡨 ... **D** ...  13 ENDIF  14 ENDFOR  15 ENDWHILE  Write the pseudocode for A, B, C and D in the algorithm. | [4] |

**(OPTIONAL, We did not do Decision Table in SH1)**

**The Notes for Decision table is available** [**Decision Table Notes and Worksheet.docx**](about:blank)**, you may read the notes and try this yourself.**

| **4** | The Income Tax Act for a country is dsecribed as follows :  For persons in the age group 16-65 (both inclusive) tax payment have to be calculated based on the person's annual income. A person earning less than $20,000 will pay 20% taxes, otherwise they pay 40% taxes. If the person has children he/she will receive a tax reduction of 10%. | | |
| --- | --- | --- | --- |
|  | **(a)** | Create a decision table to describe all the possible conditions and the actions. | [4] |
|  | **(b)** | Simplify your decision table by removing redundancies | [1] |
|  | **(c)** | Draw a flow chart to calculate the total amount of taxes to be paid when an input of the annual income is given by a user. | [4] |

**Q5 is a Data Structure problem. (Yes, in 2020 we covered data structures in SH1)**

**BLANK PAGE**

**BLANK PAGE**

**BLANK PAGE**