Practice Theory Questions

Question 1

A hash table has an index range of 1 to 900. The following pseudocode describes an algorithm for searching the table using the hashing function Hash. It is assumed that the key is present in the table.

```
1. Index <- Hash(Key)
2. WHILE Table[Index, 1] <> Key
3. Index <- Index + 1
4. ENDWHILE
5. Value <- Table[Index, 2]
```

(a) Explain the purpose of:

(i) line 3

(ii) line 5 [4]

(b) Describe a problem that might occur with a key which, when hashed, produces an index of 900. [2]

(c) What modification to the algorithm is required to overcome this problem? [3]

(d) Explain how a new item can be added to this hash table. [4]

Question 2

Bank customers are allowed to withdraw money from their accounts at an ATM. They cannot withdraw more than the current balance in their account. There is a daily limit on the amount that can be withdrawn. In some circumstances a charge is made for the transaction. The rules are:

- the transaction is rejected if the withdrawal amount requested is greater than the current balance
- the transaction is rejected if the withdrawal amount exceeds the daily limit
- if the current balance before the transaction is carried out is less than 50 dollars then any successful transaction incurs a fixed charge

(a) Create a decision table showing all the possible conditions and actions. [4]

(b) Simplify your decision table by removing redundancies. [4]

(c) Using your answer in (b) write a function using pseudocode. The function returns:

- -1 to indicate a rejection;
- 0 for a charge-free successful transaction;
- the charge for a chargeable successful transaction. [5]

(d) State two ways in which your answer in (c) demonstrates clarity of code. [2]

Question 3

The ASCII code for the character 'Z', expressed as a denary integer, is 90.

(a) Express the denary integer 90 as:

(i) an eight-bit binary number

(ii) a hexadecimal number [2]

(b) Give two reasons why hexadecimal numbers are used in computing. [2]

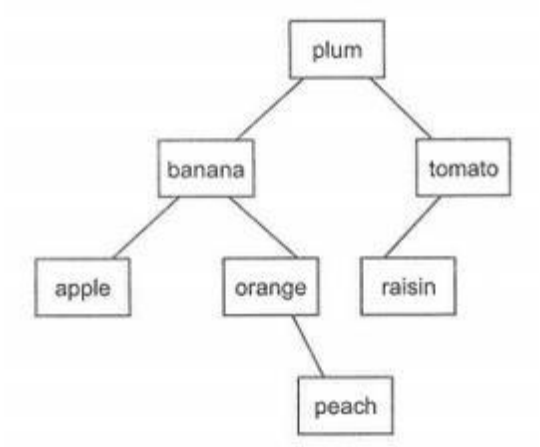
(c) State the ASCII code for 'X' in denary. Explain your answer. [2]

(d) Explain why the Unicode encoding system has replaced ASCII. [2]

(e) Describe a method of storing strings of characters of variable length in a computer. [2]

Question 4

Consider the following binary tree:



(a) List the nodes, in order, that are visited for a post-order traversal. [2]

(b) List the nodes, in order, that are visited for an inorder traversal. [2]

(c) What property is exhibited by the list of items produced in **part (b)**? [1]

(d) Describe an algorithm, using pseudocode, to perform a binary tree search. The output should state whether or not the item is present in the tree. [5]

Question 5

A function is to be written that returns the sum of all values held in an array that are greater than a minimum value. The function will be used with arrays of varying size, but never more than a maximum of 50 000 elements.

A first attempt at writing the program code for the function is given below:

```
1.  FUNCTION TotalSum(Results : ARRAY[50000] OF REAL, ArraySize : INTEGER, MinValue : REAL)
                      RETURNS REAL
2.    DECLARE Sum, Counter : INTEGER
3.    DECLARE Temp : Real
4.    Sum = 0.0
5.      FOR Counter = 1 TOO ArraySize
6.        Temp = Results[Counter]
7.        IF Temp > MinValue THEN Sum = Sum * Temp
8.      ENDEOR
9.    RETURN Sum
10. ENDFUNCTION
```

The function is included in a program specifically written to test the function. The main program outputs the value returned by the function. A compiler was used to compile the source program.

(a) The compiler reported an error at line 5 in the function. Identify the error and explain why it was flagged as a syntax error. [2]

(b) The compiler also reported an error at line 8. State the type of error reported by the compiler justifying your answer. [2]

The errors indicated in **parts (a)** and **(b)** were corrected. A successful compilation produces executable code. When the code was executed, the program failed to complete and reports an error at line 7.

(c) **(i)** State the type of error that occurred. Justify your answer. [2]

(ii) The error described in **part (c) (i)** depends on the detection of another type of error. Name this other type of error. How should the code be changed to correct this error? [2]

When the program finally runs without error, the test plan needs to be completed. The test plan uses data that tests different sizes of array, different array values and different minimum values.

The array `TempArray` is used in the main program as the array to be processed.

(d) Each element of TempArray stores a random value between 1.0 and 10.0.

(i) Explain why the function call:

```
TotalSum(TempArray, 1000, 5.0)
```

is not an appropriate black box test. [2]

(ii) Explain why the function call:

```
TotalSum(TempArray, 10, 10.5)
```

is not an appropriate white box test. [2]

(e) If each element of TempArray stores the value 1.0, state a function call that will be an appropriate black box test. Justify your answer. [3]

Question 6

A program is to be written to test an insertion sort algorithm. A top-down approach was used in the design of the program. The program, `InsertionSortTester`, has a number of parts:

- input integer values into the array
- output the initial values in the array
- output the sorted values in the array
- perform the insertion sort
- validate the values

(a) Draw a diagram, which exhibits top-down design, for the `InsertionSortTester` program. [3]

A list of data items is stored in the array `Values`. The pseudocode for the insertion sort algorithm is:

```
01  FOR i ← 2 TO ArraySize
02      Temp ← Values[i]
03      j ← i-1
04      WHILE (j > 0) AND (Values[j] > Temp)
05          Values[j+1] ← Values[j]
06          j ← j-1
07      ENDWHILE
08      Values[j+1] ← Temp
09  ENDFOR
```

(b) The sort algorithm is to be tested using the sequence of numbers: 6, 8, 2 and 1. Copy and complete the trace table given below.

Values						
[1]	[2]	[3]	[4]	i	j	Temp
6	8	2	1			

[6]

(c) Explain why this particular algorithm is inefficient for an array where the initial values are already in order. [2]

(d) Give **two** different test cases for the program. Justify your selection in each case. [4]

Question 7

An algorithm for converting a number n from denary to octal uses the three built-in functions:

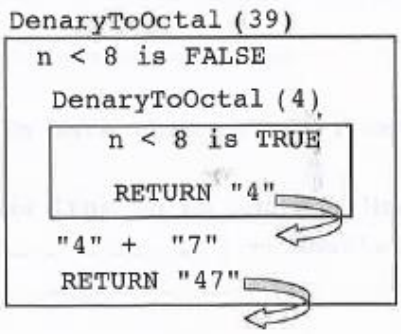
Function	Description	Example
`INTMOD(Number,Divisor)`	returns the remainder when the first parameter is divided by the second parameter.	`INTMOD(7,3)` returns 1
`INTDIV(Number,Divisor)`	returns the integer part when the first parameter is divided by the second parameter.	`INTDIV(7,3)` returns 2
`SUBSTR(ThisString,Start,Length)`	forms a substring from `ThisString`, starting at `Start` (with first index in string zero) and taking `Length` characters	`SUBSTR("abcd",1,2)` returns `"bc"`

Study the following pseudocode:

```
01  FUNCTION DenaryToOctal(n : INTEGER) RETURNS STRING
02      OctalDigits ← "01234567"
03      IF n < 8
04          THEN
05              TempString ← SUBSTR(OctalDigits,n,1)
06          ELSE
07              // '+' is the concatenation operator
08              TempString ← DenaryToOctal(INTDIV(n,8)) +
                                SUBSTR(OctalDigits,INTMOD(n,8),1))
09      ENDIF
10      RETURN TempString
11  ENDFUNCTION
```

(a) Identify where and why this is a recursive function. [2]

The diagram shows the execution of the call `DenaryToOctal(39)`.



(b) Draw a similar diagram to show the execution of the call `DenaryToOctal(67)`. [3]

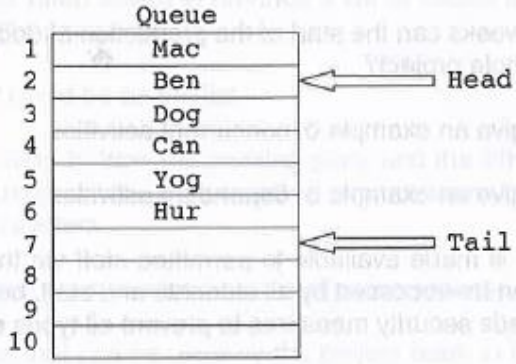
(c) Changes are to be made to the function `DenaryToOctal()` so that it converts denary numbers to hexadecimal.

Describe the changes:

- that are essential to make the revised function work.
- that are non-essential but would help with the clarity of the pseudocode. [5]

Question 8

A simple queue data structure is implemented using a one-dimensional array and two pointers, `Head` and `Tail`, as shown:



(a) Show the state of the above queue after:

- two items, Dap and Eck, are added (in that order)
- one item is removed. [3]

When ten items have been added, this simple queue cannot accept any further items.

(b) A first attempt at an algorithm for adding an item to this queue is:

```
01 IF ..........
02   THEN
03     OUTPUT "No more room to add items"
04   ELSE
05     INPUT "New item to be added", NewItem
06     Queue[..........] ← NewItem
07     ..........
08 ENDIF
```

Write the pseudocode to show the completed lines 01, 06, and 07. [3]

(c) Give the initial value for `Tail` when the queue is created and justify your answer. [2]

The programmer can reuse the space released after removing an item. This maximises the available space.

(d) Describe how the algorithm for adding an item should be amended so that the released space is made available. [2]

Question 9

A real-estate management company owns a number of residential and business units. When the company first acquires a unit, it often requires renovation work. The company records the renovation cost.

A residential unit will be a house or an individual flat within a building. A business unit will be either an office building, a storage unit or a factory.

A residential unit is either advertised for sale, with the company looking to make a profit, or retained for rental. If the unit is sold, the sale price is recorded. If the unit is retained, the monthly rental charged, the start date and length of the rental (in months) are recorded.

A business unit has a long term lease, which is usually 10 years or longer. The company records the nature of the business. It does not offer any of its business units for sale.

Other data recorded for a unit include: purchase price, purchase date, number of rooms, floor space, whether or not a lift is present. The company records whether the house has a garage and whether it has a garden.

A programmer will develop an application, using object-oriented programming to store and process the company's data.

(a) Draw a class diagram, with base class `UNIT`, showing:

- appropriate sub-class(es)
- inheritance
- the properties required
- appropriate methods, including **one** pair of 'get' and 'set' methods for **one** of the properties. [8]

(b) The company has recently purchased a number of units that they want to renovate as a 'block of flats' (a number of self-contained flats in the same building). Once the renovation is complete, the company may offer a block of flats for sale. Alternatively, it may retain the unit and advertise each individual flat for rental.

Explain how this would affect the design in **part (a)**. [3]

(c) **(i)** Explain the meaning of the term encapsulation. [2]

(ii) Explain the meaning of the term polymorphism. [2]

Question 10

A programmer implements a linked list of surnames with a start pointer, `StartPtr` and two one-dimensional arrays:

- Array `Data` stores the surnames.
- Array `Ptr` stores the link pointers.
- Both arrays have lower bound 1 and upper bound 3000.

The purpose of procedure `InsertListItem` is to insert a new surname to the linked list.

Assume a function `NextFree()` is available and returns:

- the index position for the array `Data` at which the new surname is to be inserted
- –1 when the `Data` array is full.

The programmer designs the algorithm as follows:

```
01  PROCEDURE InsertListItem(NewSurname : STRING)
02    IF NextFree() = -1
03      THEN
04        OUTPUT "List is full"
05      ELSE
06        // input the surname
07        IF StartPtr = 0
08          THEN
09            StartPtr ← NextFree()
10            Data[StartPtr] ← NewSurname
11          ELSE
12            // traverse the linked list to find the position
13            // at which to insert NewSurname
          /
          \
        ENDIF
    ENDIF
  ENDPROCEDURE
```

(a) Describe the state of the linked list, if the condition `StartPtr = 0` in line `07` is `True`. [1]

(b) It is now necessary to complete the design for procedure `InsertListItem`.

(i) The pseudocode already uses some variables.

Copy the table below and complete it to show any extra variables that you will need to use.

Variable	Data Type	Description

[3]

(ii) Write the pseudocode for line `14` onwards to complete the procedure. [6]

Question 11

The recursive procedure X has two parameters, Value and Index. The procedure processes the contents of an array, T.

```
01  PROCEDURE X(Value, Index)
02    IF T[Index] > 0
03      THEN
04        IF T[Index] > Value
05          THEN
06            X(Value, Index * 2)
07        ENDIF
08        IF T[Index] < Value
09          THEN
10            X(Value, Index * 2 + 1)
11        ENDIF
12        IF T[Index] = Value
13          THEN
14            OUTPUT "True"
15        ENDIF
16    ENDIF
17  ENDPROCEDURE
```

(a) **(i)** State what is meant by a recursive procedure. [1]

(ii) Give the **two** line numbers which indicate that procedure X is recursive. [1]

(b) An array T is used to store the data for a binary tree. A program places items in the array in the order in which they joined the tree structure.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
T	17	11	19	9	12	18	23	0	4	0	0	0	0	0	0

(i) Draw the binary tree for the array T dataset. [3]

(ii) Copy and then complete the trace table for the procedure call X(18, 1).

Procedure call	Value	Index	Output
1	18	1	

[3]

(iii) Describe the purpose of procedure X. [2]