1.1 Programming part 3

Name: _____

Class: _____

Date: _____

Time: **376 minutes**

Marks: **251 marks**

Comments:

Q1.

The following code is part of a high level program

```
Var Name: String;
Var Hours: Integer;
Var RateOfPay: Real;
    *
    *
    *
Function CalculatePay(InHours: Integer; InRateOfPay: Real): Real;
 Var Total: Real;
    *
    *
    *
    *
```

(a) Procedures and functions are often *self-contained*. What is meant by the term self-contained in this context?

_____

_____

(1)

(b) Give **one** reason why the use of global variables may introduce program bugs.

_____

_____

(1)

(Total 2 marks)

Q2.

A *recursively-defined* procedure **Process**, which takes an integer as its single parameter, is defined below.

(a) What is meant by recursively-defined?

_____

_____

(1)

(b) Describe how a stack is used in the execution of procedure **Process**?

_____

_____

_____

_____

(2)

(c) Dry run the procedure call **Process(1)**, using the data in the table below, showing clearly the order the values are printed.

```
Procedure Process (P)
  Print (P)
  If Table[P].Left <> 0
     Then Process (Table[P].Left)
  EndIf
  Print (Table[P].Data)
  If Table[P].Right <> 0
     Then Process (Table[P].Right)
  EndIf
EndProcedure
```

		Table	
	Data	Left	Right
[1]	Jones	3	2
[2]	Smith	0	0
[3]	Bremner	5	4
[4]	Fortune	0	0
[5]	Bird	0	0

Printed Output:=

_____

_____

(6)

(d) What does procedure Process describe?

_____

(1)

(Total 10 marks)

Q3.

A program has been written to store data about a school in a large town. For **each** of the following, name a suitable data type and give a reason for your choice:

(a) number of students in the school;

Data type: _____

Reason: _____

(2)

(b) the school's telephone number such as 0122456789;

Data type: _____

Reason: _____

(2)

(c) whether the school offers 'A' levels or not;

Data type: _____

Reason: _____

(2)

(d) the average number of students per teacher.

Data type: _____

Reason: _____

(2)

(Total 8 marks)

Q4.

The following code is part of a high level program

```
 Var S: String
Var Count: Integer
Var Size: Integer
S := "fred"
Size := Length(S)
If Size > 0
  Then
     For Count := 1 To Size Do
       ToUpper(S,Count)
  EndFor
EndIf
```


(a) By copying **one** relevant line from the above code, give an example of:

(i) variable declaration

_____

(1)

(ii) selection statement

_____

(1)

(iii) assignment statement

_____

(1)

(iv) iteration

_____

(1)

(b) The built in subroutines Length and ToUpper have been used in the above code.

Length(S) returns the number of characters in string S

ToUpper(S,n) converts the n^{th} character of S to upper case

Subroutines can be either *functions* or *procedures*.

(i) By ticking the correct boxes in the table below, indicate whether the subroutines are functions or procedures.

Subroutine	Procedure	Function
Length		
ToUpper		

(2)

(ii) What are the differences between a function and a procedure?

_____

_____

_____

(2)

(Total 8 marks)

Q5.

The following code is part of a high level language program:

```
Const MaxChars = 5;
Var
  Message : Array[1..MaxChars] Of Char;
  LastChar : Integer;
  Position : Integer;
Procedure Insert (c: Char);
  Begin
    If LastChar < MaxChars
      Then
        Begin
          Position := Position + 1;
          Message[Position] := c;
        End;
  End
Function Find (c: Char) : Integer;
  Var
     Found: Boolean;
  Begin
     Position := 0;
     Found := False;
     While (Position<LastChar) And Not Found
       Do
         Begin
           Position := Position + 1;
           If Message[Position] = c
             Then Found := True;
         End;
     If Found
       Then Find := Position
       Else Find  := 0;
  End;
```

(a) In **each** case, identify the following by copying **one** relevant statement from the code opposite:

(i) constant definition;

_____

(1)

(ii) variable declaration;

_____

(1)

(iii) assignment statement;

_____

(1)

(iv) selection statement;

_____

(1)

(v) iteration.

_____

(1)

(b) In **each** case, from the code opposite, identify a:

(i) local variable;

_____

(1)

(ii) global variable;

_____

(1)

(iii) parameter.

_____

(1)

(Total 8 marks)

Q6.

A recursively-defined procedure X with three integer parameters is defined below.
x DIV *y* calculates how many times *y* divides exactly into *x*. For example 7 DIV 3 = 2.

```
Procedure X (E,L,H)
 If L > H
    Then Print 'False'

    Else M ← (L+H) DIV 2
      If E = List[M]
```

```
                    Then Print 'True'
                    Else
                       If E < List[M]
                         Then X (E,L,M-1)
                         Else X (E,M+1,H)
                       Endif
                 Endif
            Endif
        EndProc
```

(a) What is meant by recursively-defined?

_____

(1)

(b) (i) Using the table below, dry-run the procedure call X (6502, 1, 11) applied to the integer array *List* containing the following elements:

Index	List
1	1234
2	1789
3	3125
4	4789
5	5006
6	5789
7	6502
8	7411
9	8407
10	8971
11	9053

E	L	H	M	List[M]	Printed Output

(7)

(ii) What process does procedure X describe?

_____

(2)

(Total 10 marks)

Q7.

A *recursively-defined* procedure B, which takes an integer as its single parameter, is defined below. The operators DIV and MOD perform integer arithmetic.
x DIV y calculates how many times y divides exactly into x. For example 7 DIV 3 = 2
x MOD y calculates the remainder that results. For example 7 MOD 3 = 1.

```
Procedure B (Number)
  If (Number = 0) OR (Number = 1)
    Then Print (Number)
    Else
     B (Number DIV 2)
     Print (Number MOD 2)
  EndIf
EndProcedure
```

(a) What is meant by recursively-defined?

_____

_____

(1)

(b) Why is a stack necessary to execute procedure B recursively?

_____

_____

(1)

(c) Dry run the procedure call *B(53)* showing clearly the values of the parameter and the printed output for the six calls of *B*.

Call Number	Parameter
1	53
2	26
3	13
4	
5	
6	

Printed Output: _____

(6)

(d) What process does procedure B describe? _____

(1)

(Total 9 marks)

Q8.

The following code is part of a high level program to manipulate text:

```
Var SI: String
Var S2: String
Var Ptr: Integer
Var L: String
S1 := "PAT"
S2 := ""                        {"" denotes an empty string}
For Ptr := 1 To 3 Do
    L := Copy (S1, Ptr)
    S2 := Concat (L, S2)
EndFor
If S1 = S2
    Then Print ('True')
    Else Print ('False')
EndIf
```

(a) By copying **one** relevant line from the above code, give an example of:

(i) variable declaration_____

(1)

(ii) selection statement_____

(1)

(iii) iteration_____

(1)

(b) The built-in subroutines **Copy**, **Concat** and **Print** have been used in the above code.

Copy (S, n) returns the n^{th} character of string S
example: Copy ("ABCDE",2) returns the character "B".

Concat (S1, S2) concatenates the two strings S1 and S2 and returns a single string
example: Concat ("ABCD","EF") returns the string "ABCDEF".

Print (S) prints the string S as output.

Subroutines are either *functions* or *procedures*. Indicate, by ticking the correct boxes, what each of the above subroutines is.

subroutine	**procedure**	**function**
copy		
concat		
print		

(3)

(c) Dry run the above code by completing the table below.

S1	Ptr	L	S2
"PAT"			""
	1	"P"	"P"
Printed Output:			

(8)

(Total 14 marks)

Q9.

A multi-storey car park is controlled by a computer system as follows.

For a vehicle arriving at the barrier-controlled *entrance*:

- the computer system generates an integer number at random from a set of unused numbers which identifies the vehicle to the system
- the vehicle's driver collects a ticket containing this number from a machine at the barrier
- after a short interval a barrier is raised to enable the car to enter the car park
- the computer system remembers the current date, the arrival time and the randomly generated number.

If the car park is full a sign is lit to indicate the situation and no vehicle is allowed to enter the car park.

For a vehicle arriving at the barrier-controlled *exit*:

- the ticket is presented to a machine which reads the number on the ticket
- the computer system determines the length of time the vehicle has been parked in the car park and calculates the amount to pay
- the amount to pay is displayed on the machine
- the driver inserts the correct money into the machine
- the computer system records the length of time in minutes and the amount to pay in pence
- after a short interval the barrier is raised to enable the vehicle to exit.

(a) Taking account of the technology that could be used for ticket production at the entrance barrier, describe **two** different ways for the number assigned to the ticket to be submitted to the computer system at the exit barrier. Your answer should include a reference to the relevant input/output hardware used.

1. _____

_____

_____

2. _____

_____

_____

_____

(4)

(b) Using the table below, construct an appropriate record structure for the computer system to use to record the relevant car parking details for one vehicle. Data types should be given that would be available in a third generation programming language.

Field Name	Data Type

(5)

(Total 9 marks)

Q10.

The list **Days** contains the following representation of the days of the week.

[Sun, Mon, Tue, Wed, Thu, Fri, Sat]

The table below shows some functions which take a list as their single argument and return a result which is either an element of a list, another list, or a Boolean value.

Head(list) - returns the element at the head of **list** (e.g. Head(Days)→ Sun) if **list** is non empty otherwise it reports an error.
Tail(list) - returns a new list containing all but the first element of the original list (e.g. Tail(Days) → [Mon, Tue, Wed, Thu, Fri, Sat]) if **list** is non-empty otherwise it reports an error.
Empty(list) - returns True if **list** is the empty list or False otherwise. The empty list is denoted by[]

(a) What result is returned when the following function calls are made?

(i) Head (Tail(Days))_____

(1)

(ii) Tail ([(Head(Days)])_____

(1)

(iii) Empty(Tail(Tail(Tail(Days))))_____

(1)

(b) Explain why it is faster to access these elements if the above data is stored as a one dimensional array.

_____

_____

(2)

(Total 5 marks)

Q11.

The following code is part of a high level language program to manage a telephone contact list:

```
Const Max = 200
Type TMember = Record
                 Name: String
                 TelNo: String
                 Age: Integer
               EndRecord
Var Member: Array [1..Max] Of TMember
Procedure FindTelNo (WantedName: String)
Var EndOfList: Boolean
Begin
  EndOfList := False
  Ptr:= 1
  While WantedName < > Member[Ptr].Name And Not EndOfList Do
     Ptr := Ptr + 1
     If Ptr > Max Then EndOEList : = True
  EndWhile
  If EndOfList
     Then Print ('Name not in list')
     Else Print (Member[Ptr-1].Name, 'tel: ', Member[Ptr-1].TelNo)
  Endlf
End
```

(a) Identify the following by copying **one** relevant statement from the above code.

(i) constant definition: _____

(1)

(ii) assignment statement: _____

(1)

(iii) selection statement: _____

(1)

(iv) iteration: _____

_____

(1)

(b) Identify the following by copying **one** relevant part statement from the above code.

(i) user-defined type: _____

(1)

(ii) parameter:_____

(1)

(iii) local variable: _____

(1)

(c) Why is it considered to be good programming practice to use named constants such as **Max**?

_____

_____

(1)

(d) (i) Why is it not good design to use a field **Age** when storing personal details?

_____

_____

(1)

(ii) What could the programmer have done instead?

_____

_____

(1)

(e) What values can a Boolean expression take?

_____

(1)

(Total 11 marks)

Q12.

(a) Machine code is the first generation of programming languages. All other generations of programming languages need a program translator before the program can be executed. Name a type of translator suitable for:

(i) Second generation language programs:

_____

(1)

(ii) Third generation language programs:

_____

(1)

(b) Imperative *high level languages* are third generation.

Give **two** characteristics of high level languages that distinguish them from second generation languages.

1. _____

2. _____

(2)

(c) In one high level language an example of a constant definition would be

CONST VatRate = 17.5;

State **one** advantage of using a named constant, like VatRate, rather than the actual value (17.5) in a high level language program.

_____

_____

(1)

(d) (i) Name an imperative high level language which you have studied.

_____

(1)

For the language you have named in (d) (i) above, give an example, using the correct syntax, of:

(ii) iteration: _____

_____

_____

(2)

(iii) selection: _____

_____

_____

(2)

(Total 10 marks)

Q13.

A program has been written to analyse the results of a survey. For **each** of the following, name a suitable data type and give a reason for your choice:

(a) number of cars owned by a household;

data type _____

(1)

reason: _____

(1)

(b) a telephone number such as 0122456789;

data type _____

(1)

reason: _____

(1)

(c) whether a household's accommodation has central heating or not,

data type _____

(1)

reason: _____

(1)

(d) the average number of cars owned by the households.

data type _____

(1)

reason: _____

(1)

(Total 8 marks)

Q14.

The structured approach when writing programs uses functions and procedures.

(a) Give two reasons why procedures are used.

1. _____

_____

2. _____

_____

(2)

(b) What are parameters used for in the context of procedures and functions?

_____

_____

(1)

(Total 3 marks)

Q15.

The following code is part of a high level language program:

```
CONST Max = 5;
VAR Tptr : INTEGER;
VAR Store : ARRAY[1..Max] OF CHAR;
PROCEDURE add (a: CHAR);
BEGIN
  IF Tptr < Max THEN
  BEGIN
     Tptr := Tptr + 1;
     Store[Tptr] :=a;
  END;
END;
FUNCTION Take : CHAR;
VAR Ptr: INTEGER;
BEGIN
```

```
        IF Tptr>0 THEN
        BEGIN
          Take := Store[1];
          Tptr := Tptr-1;
          FOR Ptr := 1 TO Tptr DO store[Ptr] := store[Ptr+1]
        END;
    END;
```

Identify the following by copying **one** relevant statement from the above code.

(a) constant definition_____

(1)

(b) variable declaration _____

(1)

(c) local variable _____

(1)

(d) global variable _____

(1)

(e) parameter_____

(1)

(f) assignment statement _____

(1)

(g) selection statement _____

(1)

(h) iteration_____

(1)

(Total 8 marks)

Q16.

The list Ports contains the following names:

[Southampton, Barcelona, Athens, Alexandria, Tunis, Lisbon]

The table below shows some functions which take a list as their single argument and return a result which is either an element of a list or a boolean value.

Head(list) – If the list is non-empty, it returns the element at the head of the list (e.g. Head (Ports) → Southampton) otherwise it reports an error
Tail(list) – If the list is non-empty it returns a new list containing all but the first element of the original list, otherwise it reports an error
Empty(list) – if the list is the empty list it returns True otherwise it returns False. The empty list is donated by [ ]

(a) What result is returned when the following function calls are made?

(i) Tail(Ports) _____

_____

(1)

(ii) Head(Tail(Tail(Ports))) _____

_____

(2)

(iii) Empty(Tail(Tail(Tail(Tail(Tail(Tail(Ports)))))) _____

_____

(2)

A *recursively defined* procedure P, which takes a list as its single parameter, is defined below.

```
Define Procedure P(list)
     If Not Empty(list)
       Then
         P(Tail(list))
         Print Head(list)
     EndIf
EndDefine
```

(b) What is meant by recursively defined?

_____

_____

(1)

(c) Explain why a stack is needed to execute procedure P recursively.

_____

_____

_____

(2)

(d) For the procedure call P(Ports) give the PRINTed output in the order in which it is produced.

_____

_____

(4)

(e) Complete the table to show the list Ports as a linked list so that the ports can be accessed in alphabetical order.

1	Southampton	
2	Barcelona	
3	Athens	
4	Alexandria	
5	Tunis	
6	Lisbon	

Head Pointer

(2)

(Total 14 marks)

Q17.

(a) Programmers are encouraged to adopt a structured approach to writing programs. One reason is so that programmers can write code which can be more easily understood by another programmer.

Explain **two** other reasons.

1. _____

_____

(2)

2. _____

_____

(2)

(b) Give **three** features of an imperative high level programming language which allow programmers to write "easy-to-understand" code.

1. _____

_____

2. _____

_____

3. _____

_____

(3)

(c) Distinguish between a compiler and an interpreter.

_____

_____

_____

_____

(2)

(Total 9 marks)

Q18.

For **each** of the following, name a suitable data type and give a reason for your choice.

(a) The number of books a student has out on loan from the college library.

Data type_____

(1)

Reason_____

(1)

(b) A student's average mark to two decimal places.

Data type_____

(1)

Reason_____

(1)

(c) A telephone number such as 01223123456.

Data type_____

(1)

Reason_____

(1)

(Total 6 marks)

Q19.

(a) An example of an iteration in Pascal is:

FOR x : = 1 TO 10 DO writeln ('Hello');

In a high level programming language you are familiar with, using the correct syntax, give an example of:

(i) declaration; _____

(2)

(ii) assignment; _____

(1)

(iii) selection. _____

_____

(2)

(b) A one-dimensional array q contains the following characters:

q	
D	[5]
K	[4]
C	[3]
T	[2]
M	[1]

(i) Dry run the following algorithm, recording your results in the diagram.

```
FOR pointer ← 1 to 5
    s[pointer] ← q[pointer]
END FOR
pointer1 ← 1
pointer2 ← 5
REPEAT
    q[pointer1] ←s[pointer2]
    pointer1 ← pointer1 + 1
    pointer2 ← pointer2 - 1
UNTIL pointer2 = 0
```

q		s		q	
D	[5]		[5]		[5]
K	[4]		[4]		[4]
C	[3]		[3]		[3]
T	[2]		[2]		[2]
M	[1]		[1]		[1]

(10)

(ii) What is the purpose of the above algorithm?

_____

(1)

(Total 16 marks)

Q20.

(a) (i) What is meant by a *parameter* of a procedure?

_____

_____

(1)

(ii) What is meant by a *global variable*?

_____

_____

(1)

(b) When writing a procedure, why might a programmer prefer to use parameters rather than global variables?

_____

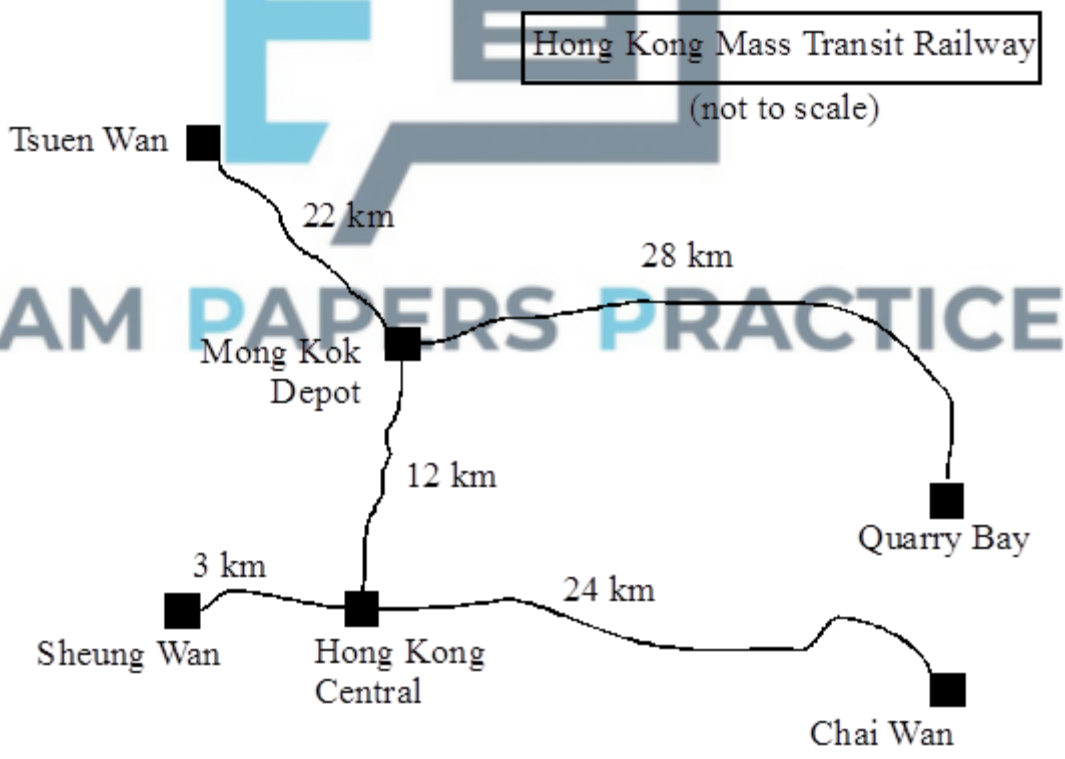
_____

(1)

(Total 3 marks)

Q21.

The plan below shows the layout of the Mass Transit Railway (MTR) in Hong Kong. The maintenance depot is at Mong Kok.



All the trains operate the same cycle (sequence) of journeys, given by the algorithm below. The algorithm is intended to ensure that:

1. trains are serviced as soon as possible after covering 135 km, and

2. each train will have travelled in both directions along each track at least once in the cycle.

The algorithm relates to three arrays called *station*, *journey* and *km*. The contents of these arrays are shown below.

Subscript	**Station**
0	Mong Kok depot (MK)
1	Tsuen Wan (TW)
2	Quarry Bay (QB)
3	Sheung Wan (SW)
4	Chai Wan (CW)
5	Hong Kong Central (HK)

Subscript	**Journey**
0	3
1	4
2	3
3	1
4	5
5	2
6	3

The 6 × 6 two-dimensional array **km**, representing the distance between stations (in kilometres), contains

		First subscript					
	km	0	1	2	3	4	5
Second Subscript	0	0	22	28	15	36	12
	1	22	0	50	37	58	34
	2	28	50	0	43	64	40
	3	15	37	43	0	27	3
	4	36	58	64	27	0	24
	5	12	34	40	3	24	0

The proposed algorithm is:

```
org:=0
last := 1
dest:= 3
maintain := FALSE
start := station[org]
finish := station[dest]
totalkm := km [org, dest]
org := dest
while(TRUE)
     n := 0
     repeat

          n := n + 1
          if (maintain = TRUE) then

               n := last
               totalkm := 0
               maintain := FALSE

          endif

          dest := journey [n]
          if (totalkm > 135) then

               dest := 0
               last := n
               maintain := TRUE

          endif

          start := station[org]
          finish := station[dest]
          totalkm := totalkm + km[org, dest]
          org := dest

     until n >= 6
endwhile
```


(a) What is the effect of the instructions **while**(TRUE) and **endwhile**?

_____

_____

(1)

(b) For each of the variables *maintain* and *n*, state with a reason what data type it should be.

_____

_____

_____

_____

_____

_____

_____

_____

(4)

(c) Copy and complete the trace table below, for one iteration of the outer (**while** .... **endwhile**) loop.

n	org	dest	last	Start	Finish	Totalkm	maintain	*comments*
	0							
			1					
		3					FALSE	
				MK				
					SW			
						15		
	3							

(8)

(d) An objective of the algorithm is that each train has travelled in both directions along every track at least once in the cycle. Using your trace table, state, with reasons, whether this objective has been achieved.

_____

_____

_____

_____

(2)

(Total 15 marks)

Q22.

What would be the result of performing **each** of the following logical operations?

(a) NOT 01001111

_____

(1)

(b) 00110000 OR 00000010

_____

(1)

(c) 11000001 AND 00010011

_____

(1)

(d) 00000101 XOR 10001110

_____

(1)

(Total 4 marks)

Q23.

The diagram below shows two masking operations.

Test	0	1	1	0	1	1	1	0	Set	0	1	1	1	0	0	0	0
Mask 1	0	0	0	0	1	0	0	0	Mask 2	0	0	0	0	0	0	0	1
Result 1									Result 2	0	1	1	1	0	0	0	1

(a) Give the content of Result 1 when Mask 1 is applied to Test using the AND operation.

_____

(1)

(b) What logical operation must be used to produce Result 2 when Mask 2 is applied to Set?

_____

(1)

(Total 2 marks)

Q24.

Using a High Level Language with which you are familiar, give **one** example of **each of** the following.

(a) Data assignment_____

_____

(b) Selection (Conditional step)_____

_____

(c) Repetition (Iteration)_____

_____

(Total 3 marks)

Q25.

A computer program contains the following fragment:

```
Boolean: LY
Integer: Y
…
LY := (Y mod 4 = 0)  AND ( NOT (Y mod 100 = 0) OR (Y mod 400 = 0) )
```

(The function A **mod** B returns the remainder when A is divided by B, eg. 11 **mod** 3 returns the value 2.)

(a) What is meant by a Boolean variable?

_____

_____

(1)

(b) What value would be assigned to the variable LY, if the variable Y contains

(i) 1999?

_____

(1)

(ii) 2000?

_____

(1)

(Total 3 marks)

Q26.

The operating system of a computer network includes the following functions and procedures:

OpenScreen(ComputerName, Channel)

where ComputerName is a character string identifying a computer on the network, and Channel is an integer identifying a communication channel.

This function opens a communication channel to the screen of the computer specified,and returns an integer, which is 0 if the function is successful, otherwise it returns one of various error codes.

SendCharacter(Char, Channel, x, y)

where Char is a character, Channel identifies the communication channel, and the integer variables x and y are screen coordinates. This procedure sends a character to the screen of the other computer using the communication channel. It does not return a value.

CloseScreen(Channel) closes the specified communication channel. It does not return a value.

InputText(Buffer) accepts a string of characters from the keyboard, terminated by a carriage return (character code 13), and stores it in Buffer. It does not return a value.

A computer on the network is running a program, designed to enable the user to send messages to another computer user. Part of the program uses the following algorithm:

```
Array of characters: Msg[50]
Integer: Count, Err, Col, Row
Character: Ch

InputText ( Msg )        // uses carriage return, code 13,  as terminator
Count := 0
Err := OpenScreen( "Admin_Computer", 10 )
if ( Err = 0 ) then
       Col := 1
       Row := 12
       Ch := "A"
       while ( Ch does not contain the code 13 ) do
              Ch := Msg[ Count ]
              SendCharacter( Ch, 10, Col, Row )
              Count := Count + 1
              Col := Col + 1
       endwhile
       CloseScreen( 10 )
else
       case ( Err ) of
              when 1: print( "Specified computer is offline
              or does not exist" )
              when 2: print( "Cannot output - network interface problem"
)
              when 3: print( "The network is down" )
       endcase
endif
```


(a) What is meant by the term parameters? Illustrate your answer by using examples of the use of parameters from the algorithm above.

_____

_____

_____

_____

_____

_____

(3)

(b) What is the benefit to the programmer of using parameters?

_____

(2)

(c) How would the array Msg be stored?

(1)

(d) Describe in detail the operation of the **while** ... **endwhile** section of the algorithm.

(4)

(e) What is the effect of the **case** … **of** … **endcase** section of the algorithm?

(2)

(f) The algorithm does not impose any limit on the length of the string the user inputs. What might happen if a string 60 characters long were entered?

(1)

(Total 13 marks)

Q27.

A routine for manipulating text strings uses the following pre-defined functions:

Len(q) returns the number of characters in the text string q;

Right$(q,p) returns a string consisting of the last (rightmost) p characters of the string q;

Left$(q,p) returns a string consisting of the first (leftmost) p characters of the string q.

The algorithm for this routine is shown below.

string: message, newstring

```
// main program

      input message
      newstring : = ""
      output message
      docharacter(message,newstring)
      output newstring
// end of program
```

procedure docharacter(a,outstring) // *both parameters passed by reference* string: piece integer: x

```
      x : = Len(a)
      piece : = Right$(a,1)
      outstring : = outstring + piece
      x : = x - 1
      if x > 0 then
            a : = Left$(a,x)
            docharacter(a,outstring)
      endif
endproc
```

Trace the algorithm and show what is output if the word CAT is input.

_____

_____

_____

(Total 9 marks)

Q28.

A security system uses the control port shown below.

7	6	5	4	3	2	1	0
0	0	0	1	1	0	0	1
Alarm	Security light	not used	Window contact	Door contact	Internal movement sensor	External movement sensor	System activated

The purpose of each bit is shown in the diagram.
Bits 0 to 4 are inputs, bit 5 is not used and should always be zero, bits 6 and 7 are output

bits.
The bit pattern shown occurs when the system is first activated.

Detection of movement results in the corresponding bit being set to 1.
Breaking of a contact results in the corresponding bit being set to 0.
Bits 6 and 7 will turn on the security light and alarm respectively when set to 1.

The system, if activated, must turn on the security light if external movement is detected.
The alarm must be turned on if either or both contacts are broken or if internal movement is detected.

(a) Give the masks and the logical operations needed for **each** of the following. In each case all other bits must remain unchanged.

(i) Testing the state of the external movement sensor.

_____

_____

_____

_____

(ii) Turning on the alarm.

_____

_____

_____

_____

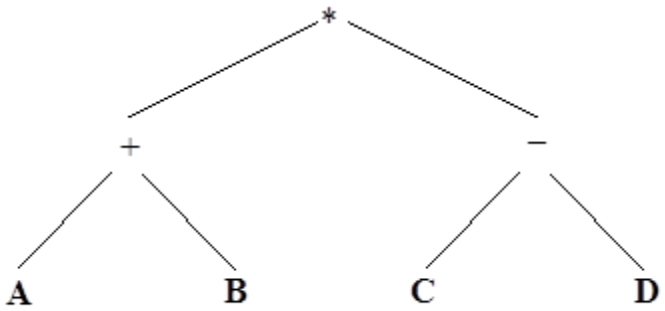
(4)

(b) Write an algorithm for the procedure required to check the sensors and contacts and to activate the alarm or security light when necessary.

(7)
(Total 11 marks)

Q29.

An algebraic expression is represented in a binary tree as follows:



(a) Label its *root*, a *branch* and a *leaf node.*

(3)

(b) Mark and label the *left sub-tree* and the *right sub-tree* of this tree.

(2)

A recursively-defined procedure T, which takes a tree structure, tree(x, y, z) as its single parameter, where x is the root, y is the left sub-tree and z is the right sub-tree, is defined below (<> means not equal to).

```
Procedure T (tree(x, y, z))

  If y <> empty
    Then
         PRINT '('
         T(y)
  EndIf
    PRINT x
    If z <> empty
    Then
         T(z)
         PRINT ')'
  EndIf
EndProc
```


(c) What is meant by *recursively-defined*?

_____

_____

(1)

(d) Explain why a stack is necessary in order to execute procedure T recursively.

_____

_____

_____

_____

_____

_____

(3)

(e) Dry run the following procedure call

```
T  (      tree( '*', tree ('+', tree ('A', empty, empty), tree ('B', empty, empty) ),
                  tree ('–', tree ('C', empty, empty), tree ('D', empty, empty) )
           )
)
```

showing clearly the PRINTed output and the values of the parameter omitted from the table (rows 4, 5, 6, 7) for the **seven** calls of T.

Call Number	Parameter
1	tree('*', tree('+', tree('A',empty,empty), tree('B',empty,empty)), tree('–', tree('C',empty,empty), tree('D',empty,empty)))
2	tree(' +', tree('A',empty,empty), tree('B',empty,empty))
3	tree('A',empty,empty)
4	
5	
6	
7	

(10)

(f) What tree traversal algorithm does procedure T describe?

_____

_____

(1)

(Total 20 marks)