ZIMBABWE SCHOOL EXAMINATIONS COUNCIL

General Certificate of Education Advanced Level

COMPUTER SCIENCE 6023/2

PAPER 2 Practical

NOVEMBER 2020 SESSION 3 hours

Additional materials:
CD for each candidate

TIME 3 hours

INFORMATION FOR CANDIDATES

This is a purely practical examination. All answers should be printed. Handwritten answers Will **not** be marked.

This paper consists of **three** sections.

Section A 20 marks

Section B 50 marks

Section C 30 marks

Answer **one** question from each section.

Each answer sheet should include the following information in the header section:

- Candidate Name and Candidate Number
- Centre Name and Date
- Subject Code

When answering programming questions, indicate the language used.

All work should be backed up by a soft copy on a CD.
If a candidate prints on more than one sheet, fasten together with string.
All answers should be correctly and clearly numbered.

This specimen paper consists of 8 printed pages

Copyright: Zimbabwe School Examinations Council, N2020.

©ZIMSEC N2020

[Turn over

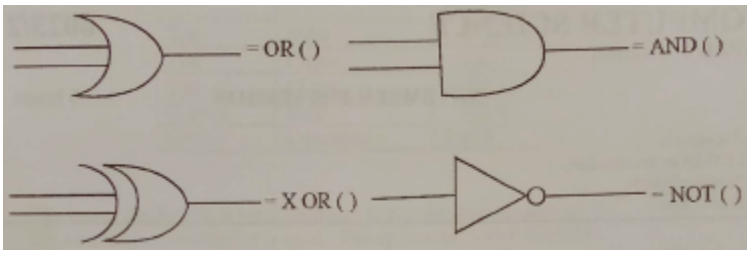
Edit with WPS Office

Section A [20 marks]

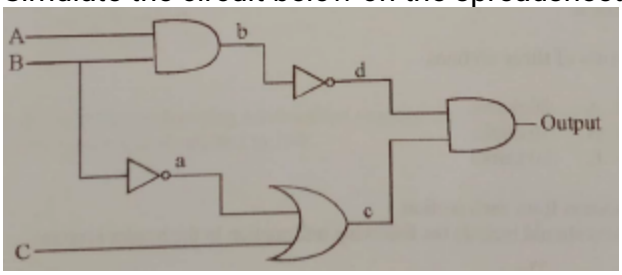
Answer **one** question from this section.

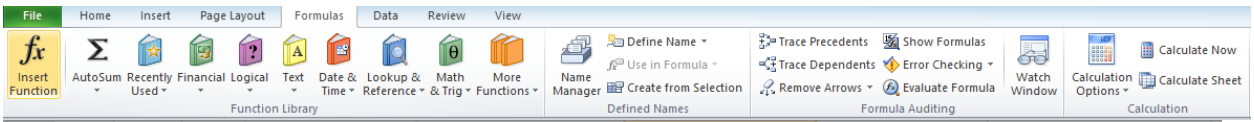
1. **Use a spreadsheet of your choice to simulate the following logic of the worksheet**

   (a) Use **only** the following formulae:



Simulate the circuit below on the spreadsheet and complete the truth table





	A	B	C	D	E	F	G	H
1	**A**	**B**	**C**	**B'=a**	**A.B=b**	**B'+C=c**	**(A.B)'=d**	**(A.B)'.(b'+c)**
2	0	0	0	=NOT(B2)	=AND(A2:B2)	=OR(C2:D2)	=NOT(E2)	**=AND(F2:G2)**
3	0	0	1	=NOT(B3)	=AND(A3:B3)	=OR(C3:D3)	=NOT(E3)	**=AND(F3:G3)**
4	0	1	0	=NOT(B4)	=AND(A4:B4)	=OR(C4:D4)	=NOT(E4)	**=AND(F4:G4)**
5	0	1	1	=NOT(B5)	=AND(A5:B5)	=OR(C5:D5)	=NOT(E5)	**=AND(F5:G5)**
6	1	0	0	=NOT(B6)	=AND(A6:B6)	=OR(C6:D6)	=NOT(E6)	**=AND(F6:G6)**
7	1	0	1	=NOT(B7)	=AND(A7:B7)	=OR(C7:D7)	=NOT(E7)	**=AND(F7:G7)**
8	1	1	0	=NOT(B8)	=AND(A8:B8)	=OR(C8:D8)	=NOT(E8)	**=AND(F8:G8)**
9	1	1	1	=NOT(B9)	=AND(A9:B9)	=OR(C9:D9)	=NOT(E9)	**=AND(F9:G9)**

	A	B	C	D	E	F	G	H
1	**A**	**B**	**C**	**B'=a**	**A.B=b**	**B'+C=c**	**(A.B)'=d**	**(A.B)'.(b'+c)**
2	0	0	0	TRUE	FALSE	TRUE	TRUE	**TRUE**
3	0	0	1	TRUE	FALSE	TRUE	TRUE	**TRUE**
4	0	1	0	FALSE	FALSE	FALSE	TRUE	**FALSE**
5	0	1	1	FALSE	FALSE	TRUE	TRUE	**TRUE**
6	1	0	0	TRUE	FALSE	TRUE	TRUE	**TRUE**
7	1	0	1	TRUE	FALSE	TRUE	TRUE	**TRUE**
8	1	1	0	FALSE	TRUE	FALSE	FALSE	**FALSE**
9	1	1	1	FALSE	TRUE	TRUE	FALSE	**FALSE**

Covert to 1s and 0s

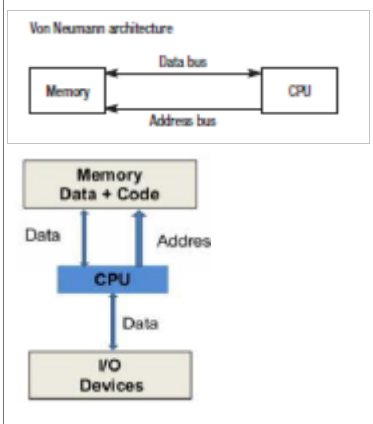
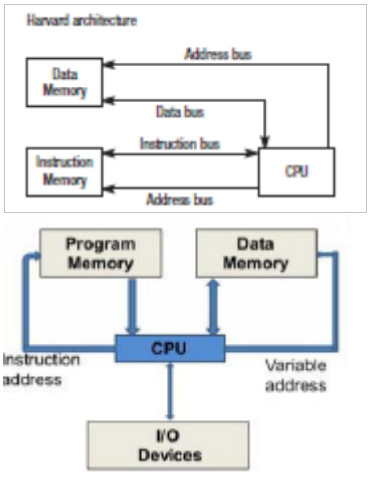
Edit with WPS Office

	A	B	C	D	E	F	G	H
1	A	B	C	B'=a	A.B=b	B'+C=c	(A.B)'=d	(A.B)'.(b'+c)
2	0	0	0	=NOT(B2)*1	=AND(A2:B2)*1	=OR(C2:D2)*1	=NOT(E2)*1	=AND(F2:G2)*1
3	0	0	1	=NOT(B3)*1	=AND(A3:B3)*1	=OR(C3:D3)*1	=NOT(E3)*1	=AND(F3:G3)*1
4	0	1	0	=NOT(B4)*1	=AND(A4:B4)*1	=OR(C4:D4)*1	=NOT(E4)*1	=AND(F4:G4)*1
5	0	1	1	=NOT(B5)*1	=AND(A5:B5)*1	=OR(C5:D5)*1	=NOT(E5)*1	=AND(F5:G5)*1
6	1	0	0	=NOT(B6)*1	=AND(A6:B6)*1	=OR(C6:D6)*1	=NOT(E6)*1	=AND(F6:G6)*1
7	1	0	1	=NOT(B7)*1	=AND(A7:B7)*1	=OR(C7:D7)*1	=NOT(E7)*1	=AND(F7:G7)*1
8	1	1	0	=NOT(B8)*1	=AND(A8:B8)*1	=OR(C8:D8)*1	=NOT(E8)*1	=AND(F8:G8)*1
9	1	1	1	=NOT(B9)*1	=AND(A9:B9)*1	=OR(C9:D9)*1	=NOT(E9)*1	=AND(F9:G9)*1

	A	B	C	D	E	F	G	H
1	A	B	C	B'=a	A.B=b	B'+C=c	(A.B)'=d	(A.B)'.(b'+c)
2	0	0	0	1	0	1	1	1
3	0	0	1	1	0	1	1	1
4	0	1	0	0	0	0	1	0
5	0	1	1	0	0	1	1	1
6	1	0	0	1	0	1	1	1
7	1	0	1	1	0	1	1	1
8	1	1	0	0	1	0	0	0
9	1	1	1	0	1	1	0	0
10								

A	B	C	B'=a	A.B=b	B'+C=c	(A.B)'=d	(A.B)'.B'.C=c.d
0	0	0	1	0	1	1	1
0	0	1	1	0	1	1	1
0	1	0	0	0	0	1	0
0	1	1	0	0	1	1	1
1	0	0	1	0	1	1	1
1	0	1	1	0	1	1	1
1	1	0	0	1	0	0	0
1	1	1	0	1	1	0	0

(b) Draw the Von Neumann and Harvard architecture explaining their differences. Justify your answer to which architecture is the best to adopt
[7]

Point of Comparison	Von Neumann Architecture	Harvard Architecture
Diagram		

Edit with WPS Office

Description	The Von Neumann architecture is a theoretical design based on the stored-program computer concept. The concept was designed by a mathematician John Von Neumann in 1945 and currently serves as the foundation of almost all modern computers.	The Harvard architecture is a modern computer architecture based on the Harvard Mark I relay-based computer model which stored instructions on punched tape (24 bits wide) and data in electro-mechanical counters. Some examples of Harvard architectures involve early computer systems where programming input could be in one media, for example, punch cards, and stored data could be in another media, for example, on tap..
Arrangement	In Von-Neumann architecture, there is no separate data and program memory. Instead, a single memory connection is given to the CPU. Has only one bus that is used for both instructions fetches and data transfers.	In Harvard architecture, the CPU is connected with both the data memory (RAM) and program memory (ROM), separately. Has separate memory space for instructions and data which physically separates signals and storage code and data memory.
Hardware requirements	In contrast to the Harvard architecture, this requires less hardware since only a common memory needs to be reached.	It requires more hardware since it will be requiring separate data and address bus for each memory.
Space requirements	Von-Neumann Architecture requires less space.	This requires more space.
Speed of execution	Speed of execution is slower since it cannot fetch the data and instructions at the same time.	The speed of execution is faster because the processor fetches data and instructions simultaneously. A program running on Harvard architecture can be executed faster and more efficiently because the instructions and data are handled more quickly as they do not have to share the same bus
Space usage	Space is not wasted because the space of the data memory can be utilized by the instructions memory and vice-versa.	It results in wastage of space since if the space is left in the data memory then the instructions memory cannot use the space of the data memory and vice-versa.
Instruction Processing	The processing unit would require two clock cycles to complete an instruction.	The processing unit can complete an instruction in one cycle if appropriate pipelining plans have been set
Cost	Instructions and data use the same bus system therefore the	Complex kind of architecture because it

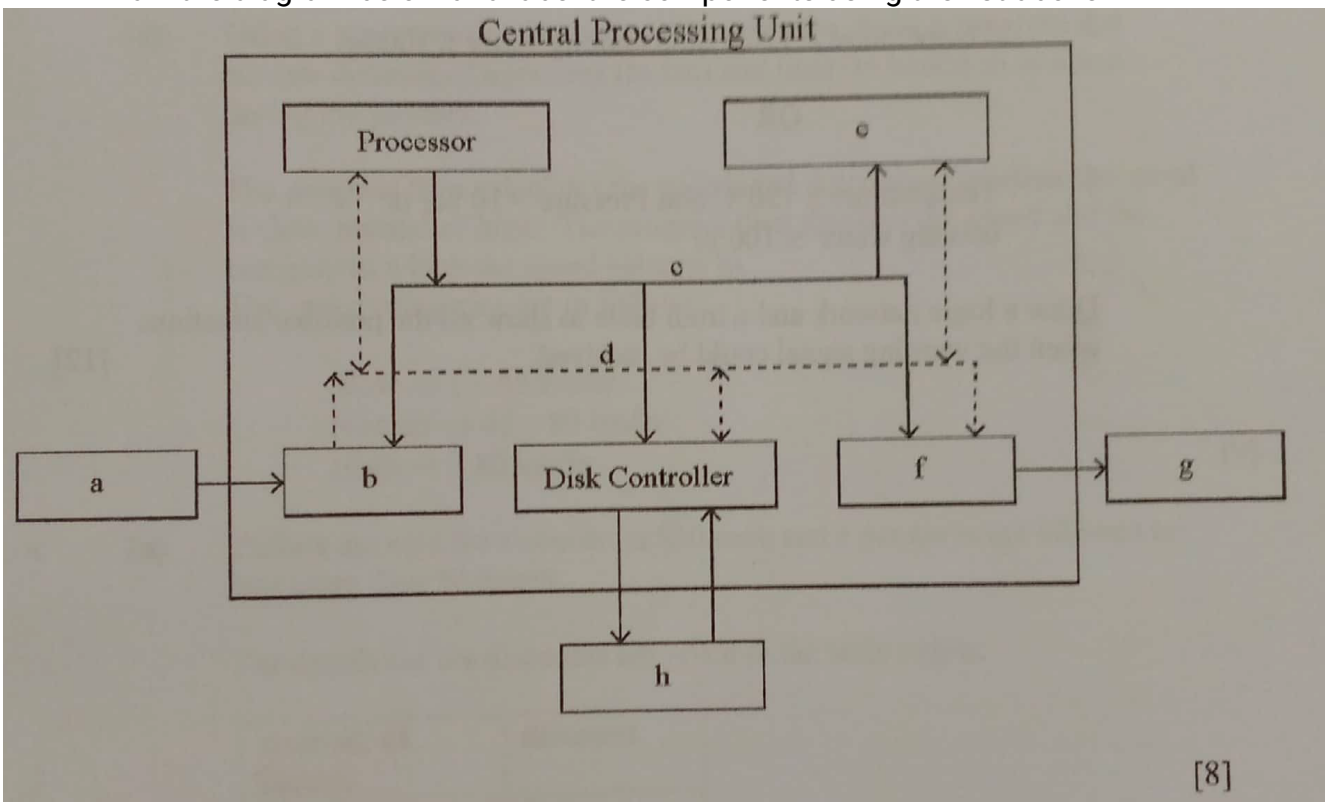
Edit with WPS Office

	design and development of control unit is simplified, hence the cost of production becomes minimum	employs two buses for instruction and data, a factor that makes development of the control unit comparatively more expensive.
Use	Von Neumann architecture is usually used literally in all machines from desktop computers, notebooks, high performance computers to workstations.	Harvard architecture is a new concept used specifically in microcontrollers and digital signal processing (DSP). . Harvard architecture is widely used on embedded computer systems such as mobile phones, burglar alarms etc.

(c) Some of the components of a computer system are:

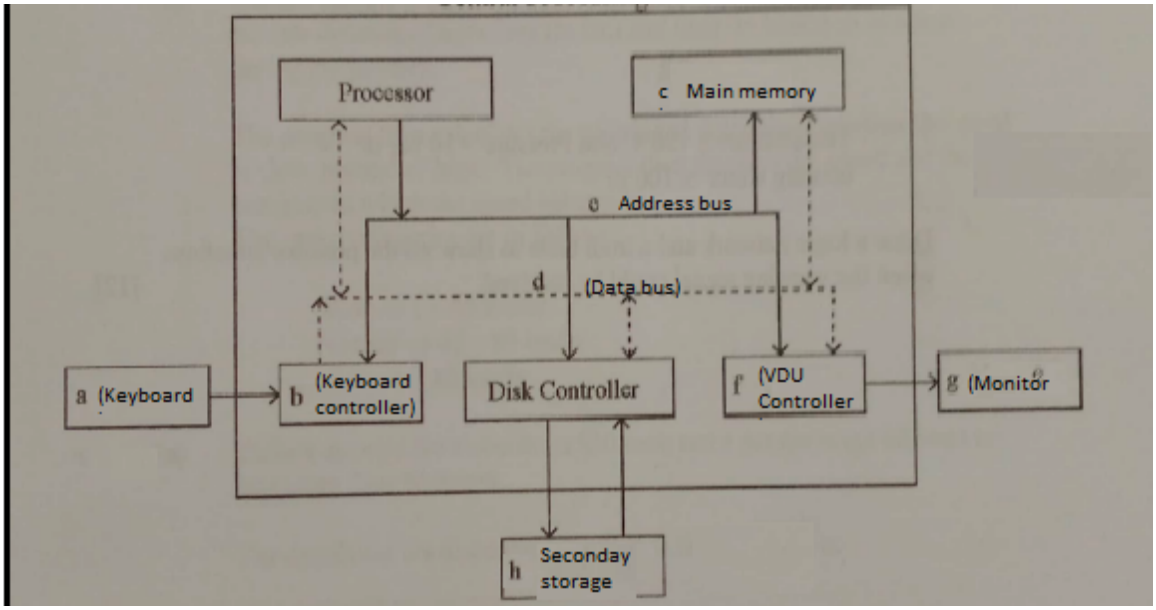
Peripherals	keyboard
	Monitor
I/O Ports	VDU controller
	Keyboard Controller
Memory	Main Memory
	Secondary Storage
System buses	Data bus
	Address bus

Draw the diagram below and label the components using the list above



[8]

Edit with WPS Office



2. (a) A retail store in Mutare wants to buy an accounting application package which will assist in carrying out daily store operations. The operations include tacking product stock, capturing sold products and printing reports for managerial staff

   As a system developer, design a suitable software architecture for mentioned requirements using an appropriate design

(b) A power station has a safety system based on three inputs to a logic network. A warning signal (S = 1) is produced when certain conditions occur based on these 3 inputs described in the table below:

Booklet 3 q10 p38

Input	Binary Value	Plant Status
T	1	Temperature > 120C
	0	Temperature ≤ 120C
P	1	Pressure > 10 bar
	0	Pressure ≤ 10 bar
W	1	Cooling Water > 100 l/hr
	0	Cooling Water ≤ 100 l/hr

A warning signal (S = 1) will be generated if:
either (a) Temperature > 120^{o}C and Cooling Water ≤ 100 l/hr
or (b) Temperature ≤ 120^{o}C and (Pressure > 10 bar or Cooling Water ≤100 l/hr)

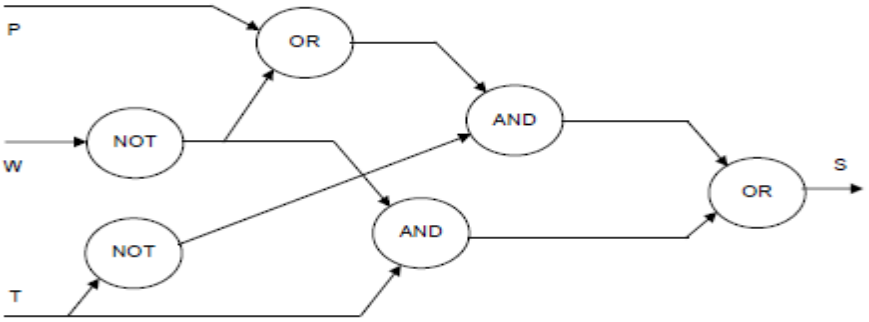
Draw a logic network and truth table to show all the possible situations when the warning signal could be received.

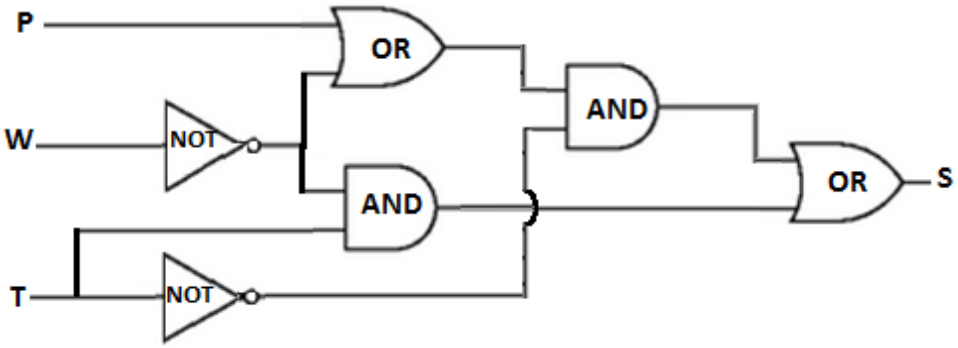
Statement: S = 1 if (T = 1 **AND** W = **NOT** 1) **OR** (T = **NOT** 1 **AND** (P = 1 **OR** W = **NOT** 1))

Expression: T•W'+T

Edit with WPS Office

Logic network:





P	W	T	W’	P+W’	W’.T	T’	(P+W’).T’	((P+W’).T’)+(W’.T)
0	0	0	1	1	0	1	1	**1**
0	0	1	1	1	1	0	0	**1**
0	1	0	0	0	0	1	0	**0**
0	1	1	0	0	0	0	0	**0**
1	0	0	1	1	0	1	1	**1**
1	0	1	1	1	1	0	0	**1**
1	1	0	0	1	0	1	1	**1**
1	1	1	0	1	0	0	0	**0**

	A	B	C	D	E	F	G	H	I
1	P	W	T	W	P+W'	W'.T	T'	(P+W').T'	((P+W').T')+(W'.T)
2	0	0	0	=NOT(B2)*1	=OR(A2;D2)*1	=AND(D2;C2)*1	=NOT(C2)*1	=AND(E2;G2)*1	=OR(H2;F2)*1
3	0	0	1	=NOT(B3)*1	=OR(A3;D3)*1	=AND(D3;C3)*1	=NOT(C3)*1	=AND(E3;G3)*1	=OR(H3;F3)*1
4	0	1	0	=NOT(B4)*1	=OR(A4;D4)*1	=AND(D4;C4)*1	=NOT(C4)*1	=AND(E4;G4)*1	=OR(H4;F4)*1
5	0	1	1	=NOT(B5)*1	=OR(A5;D5)*1	=AND(D5;C5)*1	=NOT(C5)*1	=AND(E5;G5)*1	=OR(H5;F5)*1
6	1	0	0	=NOT(B6)*1	=OR(A6;D6)*1	=AND(D6;C6)*1	=NOT(C6)*1	=AND(E6;G6)*1	=OR(H6;F6)*1
7	1	0	1	=NOT(B7)*1	=OR(A7;D7)*1	=AND(D7;C7)*1	=NOT(C7)*1	=AND(E7;G7)*1	=OR(H7;F7)*1
8	1	1	0	=NOT(B8)*1	=OR(A8;D8)*1	=AND(D8;C8)*1	=NOT(C8)*1	=AND(E8;G8)*1	=OR(H8;F8)*1
9	1	1	1	=NOT(B9)*1	=OR(A9;D9)*1	=AND(D9;C9)*1	=NOT(C9)*1	=AND(E9;G9)*1	=OR(H9;F9)*1

Edit with WPS Office

	A	B	C	D	E	F	G	H	I	J
1	P	W	T	W	P+W'	W'.T	T'	(P+W').T'	((P+W').T')+(W'.T)	
2	0	0	0	1	1	0	1	1	1	
3	0	0	1	1	1	1	0	0	1	
4	0	1	0	0	0	0	1	0	0	
5	0	1	1	0	0	0	0	0	0	
6	1	0	0	1	1	0	1	1	1	
7	1	0	1	1	1	1	0	0	1	
8	1	1	0	0	1	0	1	1	1	
9	1	1	1	0	1	0	0	0	0	

Edit with WPS Office

Section B [50 marks]

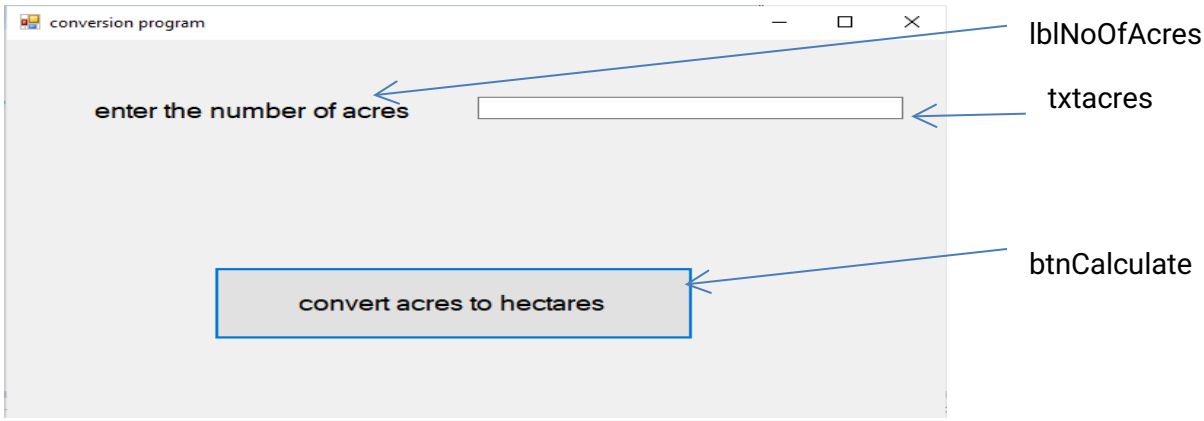
Answer **one** question from this section.

3. (a) Write an algorithm in pseudocode to calculate the average of a set of 5 marks for ten students entered through the keyboard. [8]

```
Start
    Total/Sum=0
   Average=0
    For countStudentI = 1 to 10
       Enter StuentName
         For countMark = 1 to 5
            Input/Enter/Read mark
            Total= Total+ mark ( Total +=mark)
         Next countMark
    Average=total/countMark
    Display/Print  StundentName and Average
  Next CountStudent
End/Stop
```

(b) Design an interface and write a program to convert acres to hectares. **[1 acre = 0,404686 hectares.]**
The program should accept input from user and display the output. The program should work for the real numbers as well as whole numbers (e.g. it should accept 7,.2 acres, 0.5 acres etc.)
[8]
1b visual basic 2012 used

Design



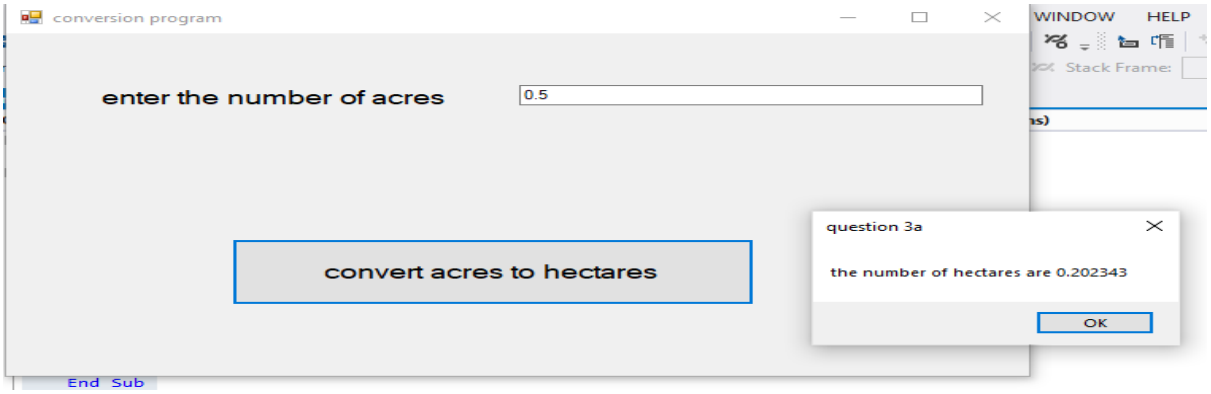
Code

```
Public Class Form1

 Private Sub btnCalculate _Click(sender As Object, e As EventArgs) Handles btnCalculate.Click
     'declaring variables'
     Dim acre As Single
     Dim hectare As Single
     Const het = 0.404686
     'assinng variable'
     acre = txtacres.Text
     'formula that will work'
     hectare = acre * het
```

Edit with WPS Office

```
    'showing the output in a massage box'
    MsgBox("the number of hectares are", "Question 3a " & hectare)
  End Sub
End Class
```

Results



(c) By using the bubble sort algorithm, write a code to sort an integer array of 10 (ten) elements in ascending order.
[25]

It works as follows,

1. Compare the first item in the list with the second.
2. If they are in the wrong order, swap them around.
3. Compare the second item to the third item.
4. If they are in the wrong order, swap them around.
5. Continue doing this until the end of the array.
6. If you have had to swap any items, repeat the process until no swaps are made.

```
For Count1 = 1 To 10
    For Count2 = 1 To 10
    If A[Count2] > A[Count2 + 1] Then
     Temp A[Count2]
     A[Count2] A[Count2 + 1]
     A[Count2 + 1] Temp
    EndIf
    Next Count2
Next Count1
```

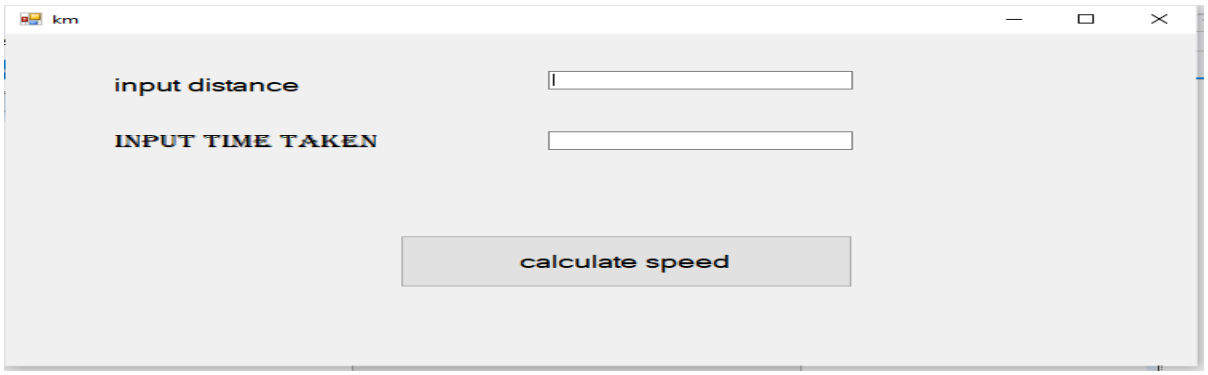
Edit with WPS Office

```
Public Sub BubbleSort(ByRef ArrayIn() As Long)
  Dim i, j    As Integer
  For i = UBound(ArrayIn) To 0 Step -1
    For j = 0 To i - 1
        If ArrayIn(j) > ArrayIn(j + 1) Then
            Call swap(ArrayIn(j), ArrayIn(j + 1))
        End If
    Next j
  Next i
End Sub

Private Sub swap(ByRef data1 As Long, ByRef data2 As Long)
  Dim temp As Long
  temp = data1
  data1 = data2
  data2 = temp
End Sub
```

(d) Using a programing language of your choice, write a program that accept distance of a journey (in km) and time (in hours) to be spend during the journey.
The program then calculates the speed; and determines whether the speed is slow, normal or high. The program then displays the speed category in which the speed belongs to.
The speed categories are as follows

Slow ➔ 1 – 45 km/h
Normal ➔ 1 – 45 km/h
High ➔ 1 – 45 km/h [9]

3d design

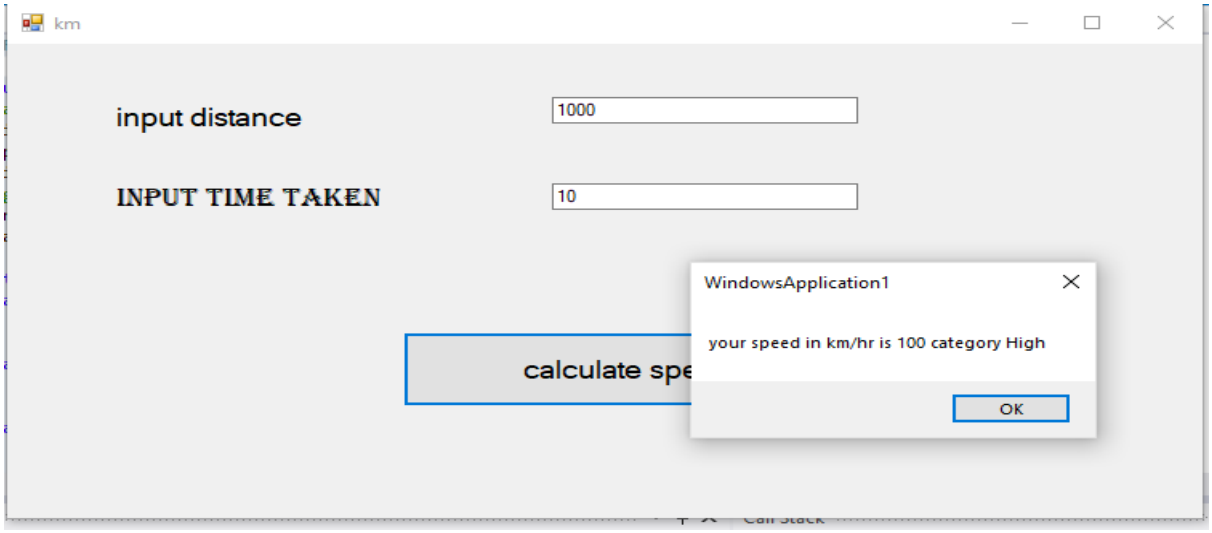


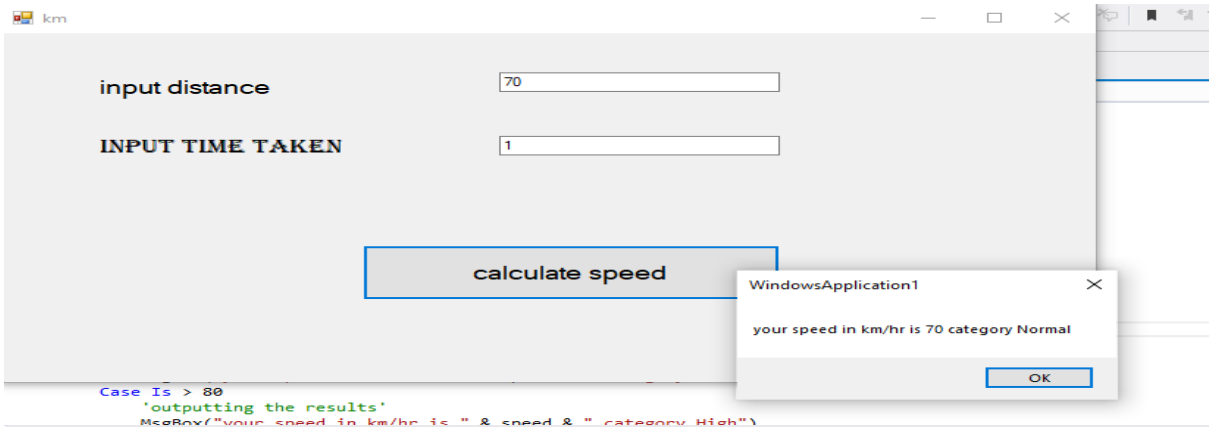
Code

```
Public Class Form1

  Private Sub btncalculate_Click(sender As Object, e As EventArgs) Handles btncalculate.Click
    'declaring variables'
    Dim distance As Single
    Dim speed As Single
    Dim timetaken As Integer
    'assigning variables'
    distance = txtdistance.Text
    timetaken = txttime.Text
    speed = distance / timetaken
    Select Case speed
      Case 1 To 45
        'outputting the results'
```

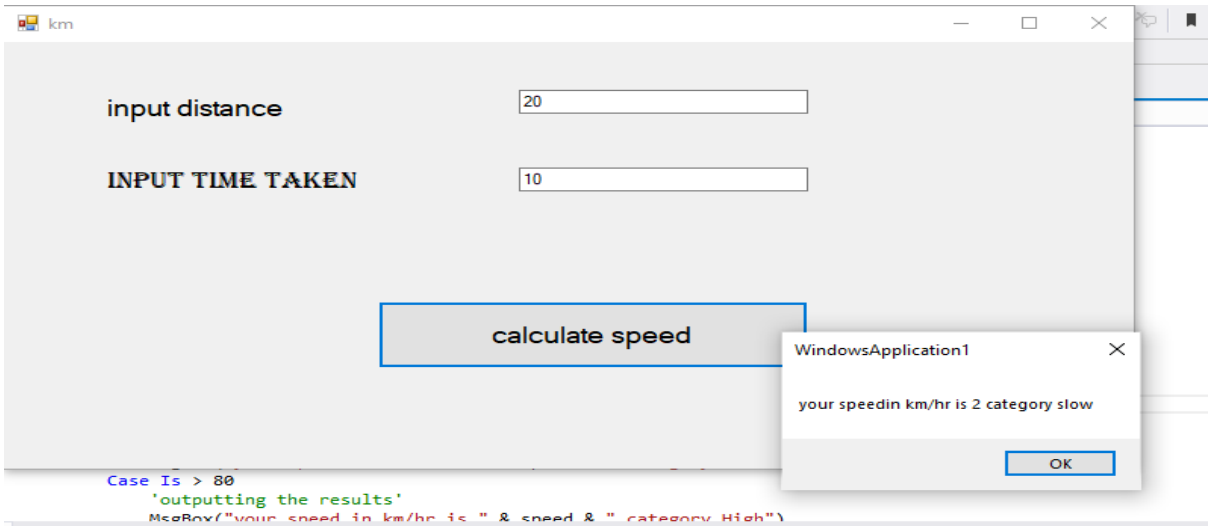
Edit with WPS Office

```
        MsgBox("your speedin km/hr is" & speed & " category slow")
    Case 45 To 80
        'outputting the results'
        MsgBox("your speed in km/hr is" & speed & " category Normal")
    Case Is > 80
        'outputting the results'
        MsgBox("your speed in km/hr is " & speed & " category High")
End Select

    End Sub
End Class
```





Edit with WPS Office



4. (a) Tickets are sold for a concert at $30 each and a person is not allowed more than 50 tickets.

The details for the discounts are given in the table below

Number of tickets	discount
10 -20	10%
21 -50	20%

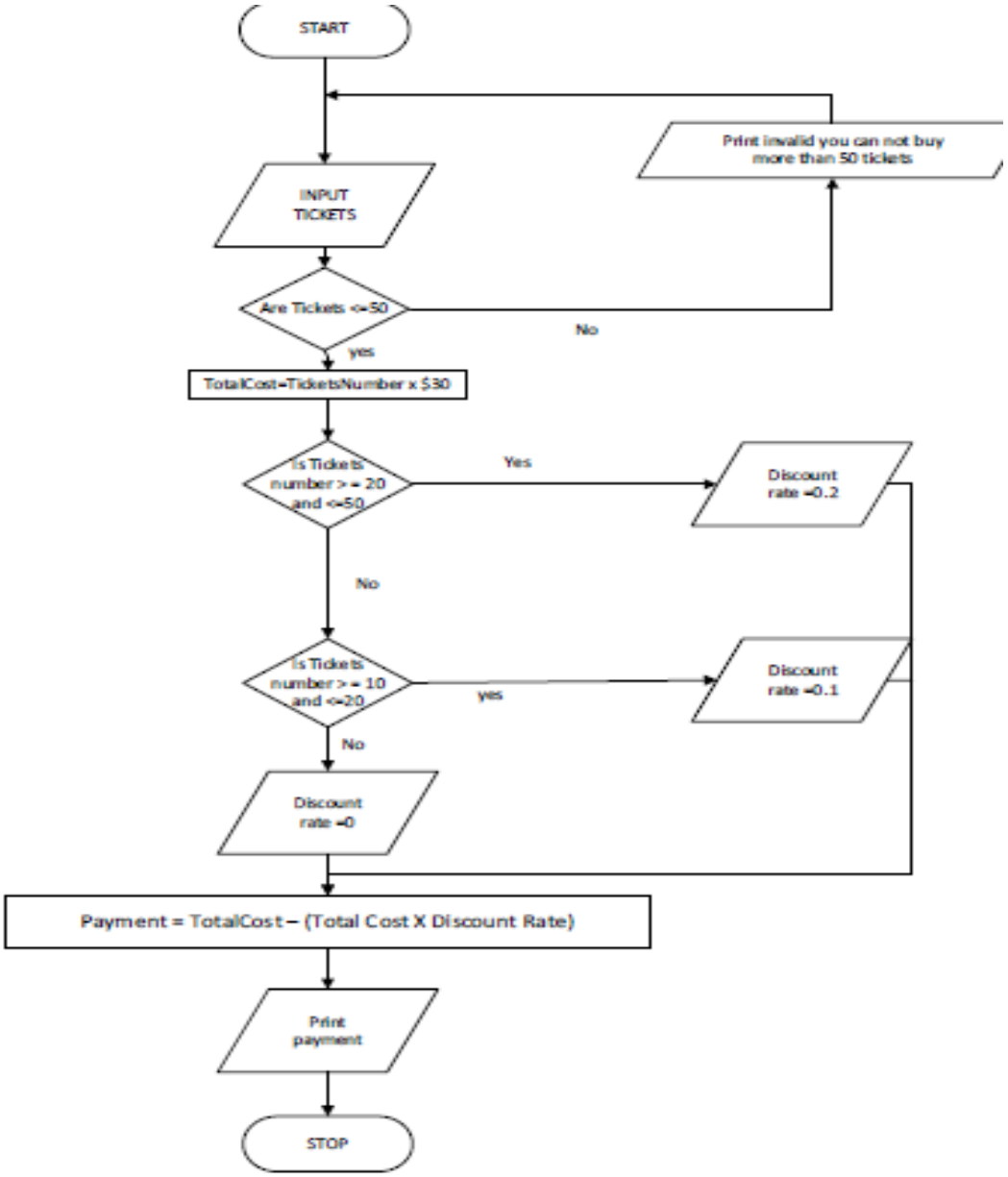
Draw a flowchart of a program that accepts the number of tickets, calculates discount and displays the total cost of the tickets. [12]

```
START
 READ TicketsBought
   ticketsCost = $30  'constant
   IF TicketsBought > 50 THEN
    OUTPUT " Invalid, Please re-enter tickets bought
   ELSE
    OUTPUT  TotalCost = Ticket Cost* TicketsBought
   END IF
   IF TicketsBought >= 10 AND TicketsBought < 20 THEN
        Discount = TotalCost *0.1
   ELSE IF TicketsBought >= 20 AND TicketsBought < = 50 THEN
        Discount = TotalCost *0.2
   ELSE
       Discount = TotalCost *0
   END IF
 Payment = TotalCost – Discount
 OUPUT Payment
STOP

REPEAT
        PRINT "How many tickets would you like to buy?"
        INPUT NumberOfTickets
UNTIL NumberOfTickets > 0 AND NumberOfTickets < 51

IF NumberOfTickets >=10 AND NumberOfTickets <21 THEN
```

Edit with WPS Office

```
        Discount ← 0.1
ELSE IF NumberOfTickets >=10 and NumberOfTickets <51 THEN
        Discount ← 0.2
ELSE
        Discount ← 0
ENDIF
    Cost ← NumberOfTickets * 30 * (1 – Discount)
    PRINT "Your tickets cost", Cost

    OR
REPEAT
        PRINT "How many tickets would you like to buy?"
        INPUT NumberOfTickets
UNTIL NumberOfTickets > 0 AND NumberOfTickets < 51

IF NumberOfTickets < 10 THEN
        Discount ← 0
ELSE IF NumberOfTickets >=10 AND NumberOfTickets <21 THEN
        Discount ← 0.1
ELSE ( IF NumberOfTickets >=10 and NumberOfTickets <51 THEN )
        Discount ← 0.2

ENDIF
    Cost ← NumberOfTickets * 30 * (1 – Discount)
    PRINT "Your tickets cost", Cost
```

Edit with WPS Office



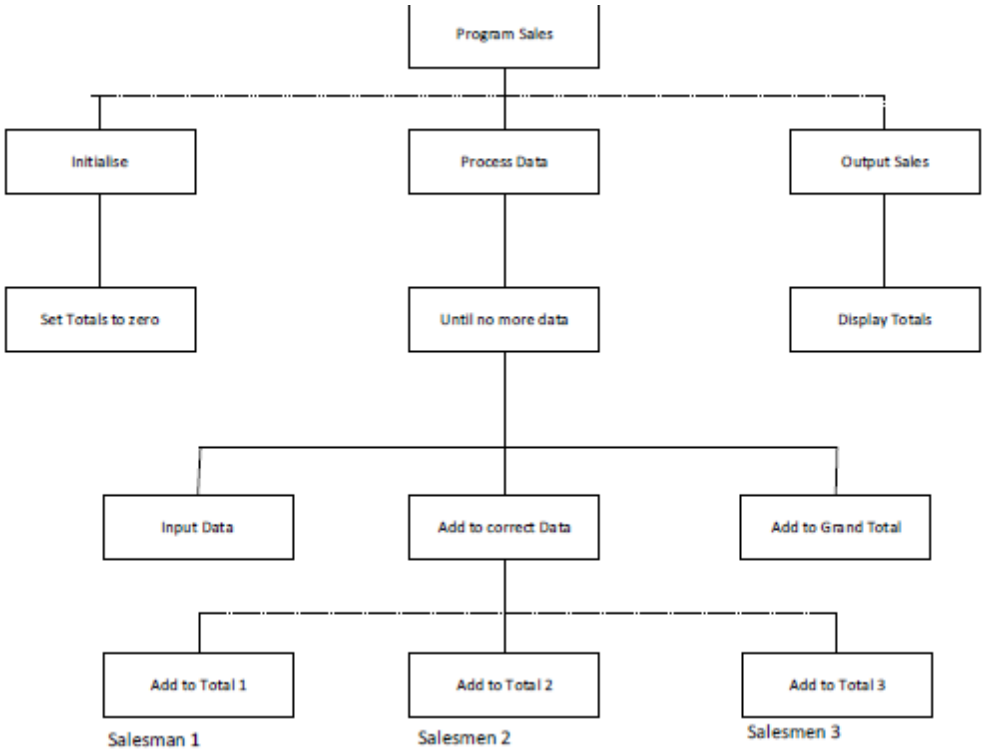
(b) A program accepts total amount of daily sales for each transactions and the grand total sales for

each sales person and the total of all the sales. The program then outputs sales person number, total

sales per salesperson and grand total.

(i) Draw a structure diagram to illustrates the program, [13]

Edit with WPS Office



(ii) Write a program for the scenario described. [25]

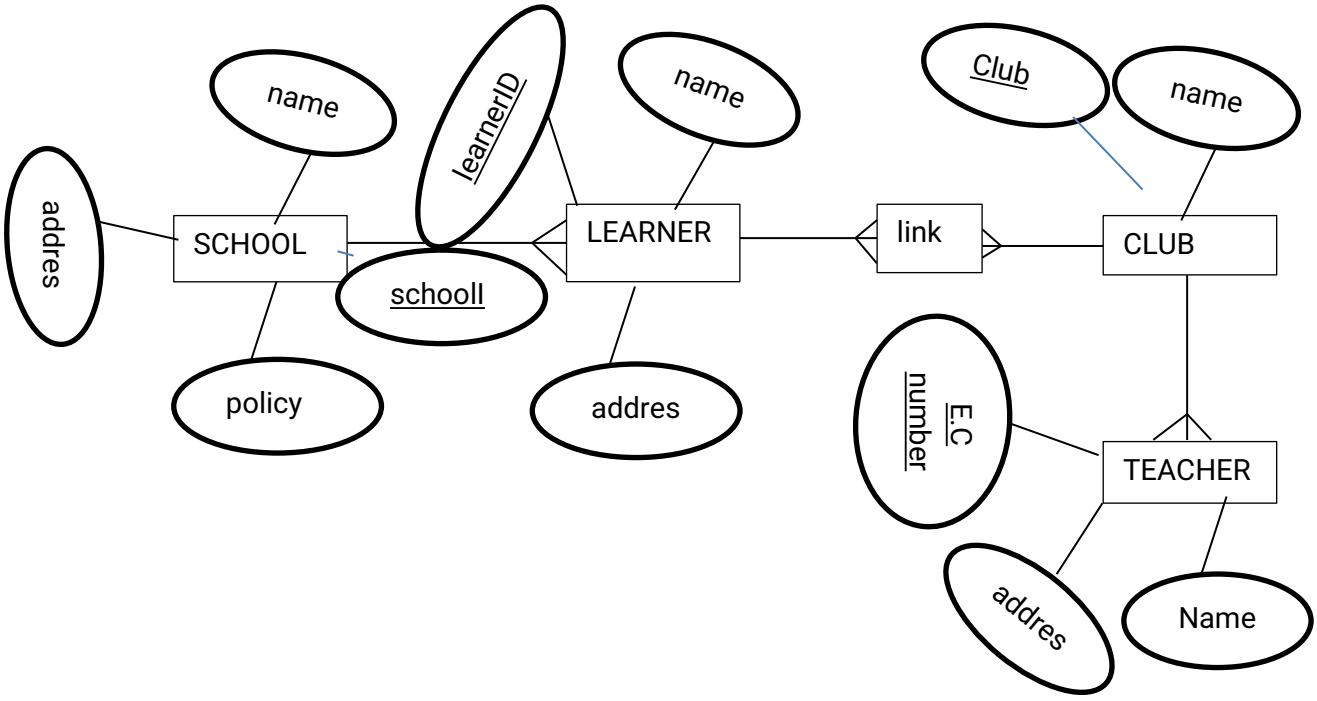
Edit with WPS Office

Section C [30 marks]

Answer **one** question from this section.

5. (a) A school has a policy that a leaner has to participate in as least one club. Each club has at least one
patron who is a teacher. A teacher may only patron one club.

(i) Draw an ER-diagram from this system with clear labels on the relationships amongst the entities in the scenario, in second normal form(2NF). [6]



6.

(ii) Suggest attributes for each entity by drawing on the same diagram or by typing in a separate
table from the database model
State the primary key for each entity.
[9]

Entity	Attributes	Primary key
School	SchoolID, Address Name, Policy	schoolID
Learner	LearnerID, Address, Name	learnerID
Club	ClubID, Name	ClubID
Teacher	EC Number, Address, Name	EC number

(b) A shopkeeper uses a single file to record details of sales of goods

Edit with WPS Office

Draw the logic circuits and complete the truth tables for the following logic statements and Boolean algebra statements:

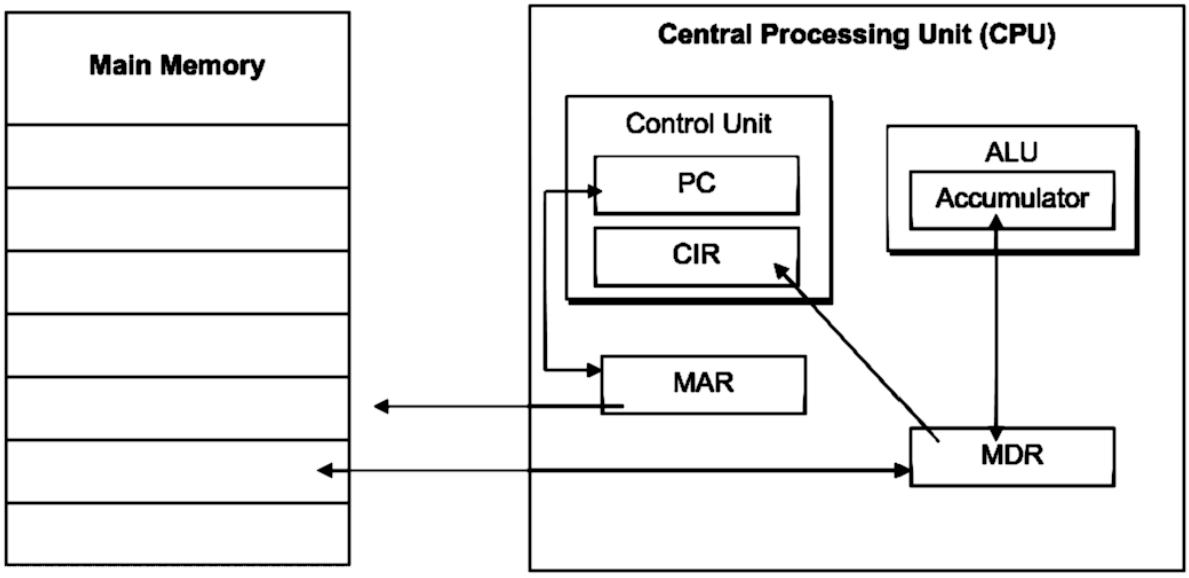
a X : 1 if{A = 1 OR B=1)0R(A=0AND B = 1)

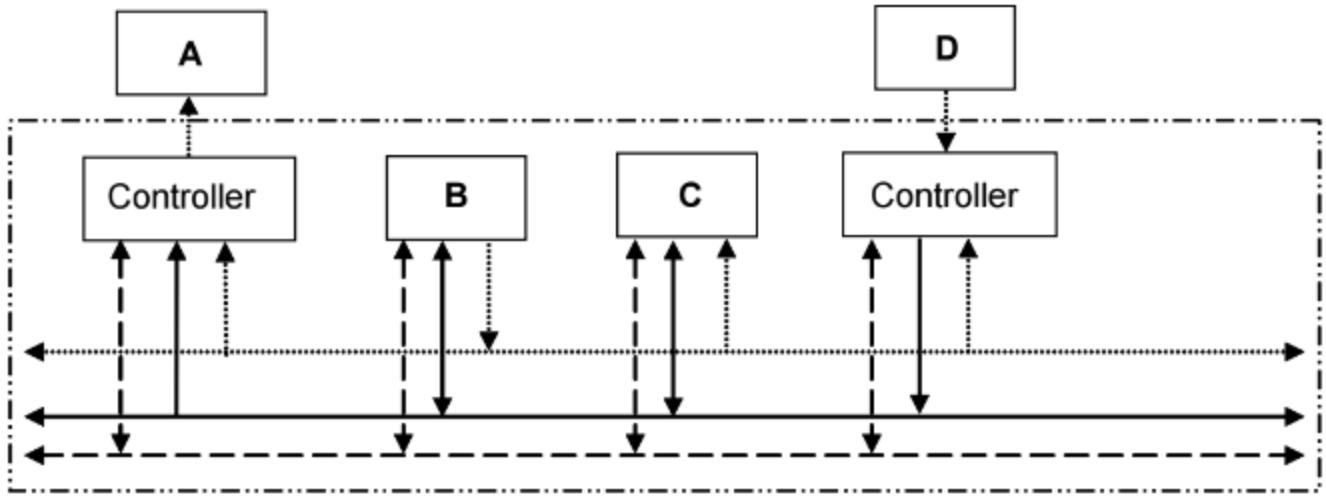
b Y = 1 if{A = O AND B = O) AND {B = 0 OR C = 1)

c T = 1 if (switch K is ON or switch L is ON) OR (switch K is ON and switch M is OFF) OR (switch M is ON)

d X:1 if (a.b)+(b.c}

a R = 1 if (switch d is A is ON and switch 8 is ON) AND (switch 8 is ON or switch C is OFF)





Write in the corresponding space below, the correct name for each of **A**, **B**, **C** and **D** from the figure above using only the following:

Edit with WPS Office

- Processor
- Address Bus
- Data Bus
- Main Memory
- Keyboard
- Visual Display Unit

Answer:

A = Visual display unit

B = Processor

C = Main memory

D = Keyboard

(b) Show the stages of a quick sort to put the following numbers into ascending numerical order:
24 8 37 11 15 (OCR 2509 January 2005 q4)

• highlight first number in the list (the 'search number')
• pointer at each end of list
• Repeat:
Compare numbers being pointed to…
…if in wrong order, swap
Move pointer of non-search number
Until pointers coincide so search number in correct position
Split list into 2 sublists
Quick sort each sublist
• Repeat until all sublists have a single number
• Put sublists back together

Quick Sort

- First the data is placed in a row with an arrow under the first and last values, pointing at each other, one is fixed where as one is movable.
- If the two values are in the correct order then move the movable arrow towards the fixed arrow, else swap the items and the arrows.
- Continue to repeat this until the arrows collide.
- Continue to repeat this process until the files are of a length of one.

Edit with WPS Office

	24 →	8	37	11	15 ←
Swap 24 & 15	15 →	8	37	11	24 ←
	15	8 →	37	11	24 ←
	15	8	37 →	11	24 ←
Swap 37 & 24	15	8	24 →	11	37 ←
	15	8	24 →	11 ←	37
Swap 24 & 11	15	8	11 →	24 ←	37
	15	8	11	24 ← →	37

Split into sublists, repeat…

OR

alternative answer using a pivot:

- **Select an item at random, the pivot**
- **Create two new lists:**
- **…one with all items less than pivot,**
- **…other with items greater than pivot**
- **Repeat**
- **…until lists only have one item**

(demonstrate this on numbers given)

24	8	37	11	15	Pivot is 37
24	8	11	15	37	numbers moved to left of pivot as smaller, choose 11 as new pivot for left section
8	11	24	15	37	24 moved to right of 11 as larger

etc

[6]

Quick sort:

Uses two pointers

Compare numbers at pointers & swap if in wrong order

Move one pointer at a time

Alternative method uses a pivot

Repeated splitting into sublists

Sublists are independent so can be sorted separately using a recursive algorithm

Sublists are reduced to a single number then put together to form sorted list.

Advantage:

Very quick for large sets of data.

Disadvantage:

Initial arrangement

Edit with WPS Office

Edit with WPS Office