|  |  |  |
| --- | --- | --- |
| Q1-  a- | **What is the difference between Actual Parameter and Formal Parameters?**  **Also, give a suitable C++ code to illustrate both** | 2 |
| b- | **Write the names of the header files to which the following belong:**  **(i) frexp() (ii) isalnum()** | 1 |
| c- | **Rewrite the following program after removing the syntactical errors (if any). Underline each correction.**  **#include [iostream.h]**  **class PAYITNOW**  **{**  **int Charge;**  **PUBLIC:**  **void Raise(){cin>>Charge;}**  **void Show{cout<<Charge;}**  **};**  **void main()**  **{**  **PAYITNOW P;**  **P.Raise();**  **Show();**  **}** | 2 |
| d- | **Find the output of the following program:**  #include <iostream.h>  struct GAME  { int Score, Bonus;};  void Play(GAME &g, int N=10)  {  g.Score++;g.Bonus+=N;  }  void main()  {  GAME G={110,50};  Play(G,10);  cout<<G.Score<<":"<<G.Bonus<<endl;  Play(G);  cout<<G.Score<<":"<<G.Bonus<<endl;  Play(G,15);  cout<<G.Score<<":"<<G.Bonus<<endl;  } | 3 |
| E | **Find the output of the following program:**  **#include <iostream.h>**  **#include <ctype.h>**    **void Encrypt(char T[])**  **{**  **for (int i=0;T[i]!='\0';i+=2)**  **if (T[i]=='A' || T[i]=='E') T[i]='#';**  **else if (islower(T[i])) T[i]=toupper(T[i]);**  **else T[i]='@';**  **}**  **void main()**  **{**  **char Text[]="SaVE EArtH";//The two words in the string Text**  **//are separated by single space**  **Encrypt(Text);**  **cout<<Text<<endl;**  **}** | 2 |
| F | **In the following program, if the value of Guess entered by the user is 65, what**  **will be the expected output(s) from the following options (i), (ii), (iii) and (iv)?**  #include <iostream.h>  #include <stdlib.h>  void main()  {  int Guess;  randomize();  cin>>Guess;  for (int I=1;I<=4;I++)  {  New=Guess+random(I);  cout<<(char)New;  }  }  (i) ABBC  (ii) ACBA  (iii) BCDA  (iv) CABD | 2 |
| Q2-  a- | **What do you understand by Data Encapsulation and Data Hiding? Also, give a suitable C++ code to illustrate both.** | 2 |
| b- | 1. **Answer the questions (i) and (ii) after going through the following class:**   **class Seminar**  **{**  **int Time;**  **public:**  **Seminar() //Function 1**  **{**  **Time=30;cout<<”Seminar starts now”<<end1;**  **}**  **void Lecture() //Function 2**  **{**  **cout<<”Lectures in the seminar on”<<end1;**  **}**  **Seminar(int Duration) //Function 3**  **{**  **Time=Duration;cout<<”Seminar starts now”<<end1;**  **}**  **~Seminar() //Function 4**  **{**  **cout<<”Vote of thanks”<<end1;**  **}**  **};**   1. In Object Oriented Programming, what is **Function 4** referred as and when does it get invoked/called? 2. In Object Oriented Programming, which concept is illustrated by **Function 1** and **Function 3** together? Write an example illustrating the calls for these functions. | 2 |
| C | **Define a class TEST in C++ with following description: 4**  **Private Members**   * 1. **TestCode of type integer**   2. **Description of type string**   3. **NoCandidate of type integer**   4. **CenterReqd (number of centers required) of type integer**   5. **A member function CALCNTR() to calculate and return the number of centers as (NoCandidates/100+1)**   **Public Members**   * **A function SCHEDULE() to allow user to enter values for TestCode, Description, NoCandidate & call function CALCNTR() to calculate the number of Centres** * **A function DISPTEST() to allow user to view the content of all the data members** | 4 |
| d- | 1. **Answer the questions (i) to (iv) based on the following:**   **class CUSTOMER**  **{**  **int Cust\_no;**  **char Cust\_Name[20];**  **protected:**  **void Register();**  **public:**  **CUSTOMER();**  **void Status();**  **};**  **class SALESMAN**  **{**  **int Salesman\_no;**  **char Salesman\_Name[20];**  **protected:**  **float Salary;**  **public:**  **SALESMAN();**  **void Enter();**  **void Show();**  **};**  **class SHOP : private CUSTOMER , public SALESMAN**  **{**  **char Voucher\_No[10];**  **char Sales\_Date[8];**  **public:**  **SHOP();**  **void Sales\_Entry();**  **void Sales\_Detail();**  **};**   1. **Write the names of data members which are accessible from objects belonging to class CUSTOMER.** 2. **Write the names of all the member functions which are accessible from objects belonging to class SALESMAN.** 3. **Write the names of all the members which are accessible from member functions of class SHOP.** 4. **How many bytes will be required by an object belonging to class SHOP?** | 4 |
| Q3-  A- | **Write a function in C++ to combine the contents of two equi-sized arrays A and B by computing their corresponding elements with the formula 2\*A[i]+3\*B[i]; where value i varies from 0 to N-1 and transfer the resultant content in the third same sized array.** | 4 |
| b- | An array S[40][30] is stored in the memory along the row with each of the element occupying 2 bytes, find out the memory location for the element S[20][10], if the Base Address of the array is 5000. | 3 |
| c- | **Write a function in C++ to perform Insert operation in dynamically allocated Queue containing names of students.** | 4 |
| d- | Write a function in C++ to find the sum of both left and right diagonal elements from a two dimensional array (matrix). | 2 |
| E | 1. **Evaluate the following postfix notation of expression:**   **True, False, AND, True, True, NOT, OR, AND** | 2 |
| Q4-  a- | **the program segment given below carefully and fill the blanks marked as Statement 1 and Statement 2 using seekp() and seekg() functions for performing the required task.**    **#include <fstream.h>**  **class Item**  **{**  **int Ino;char Item[20];**  **public:**  **//Function to search and display the content from a particular //record number**  **void Search(int );**  **//Function to modify the content of a particular record number**  **void Modify(int); }; void Item::Search(int RecNo)**  **{**  **fstream File;**  **File.open(“STOCK.DAT”,ios::binary|ios::in);**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //Statement 1**  **File.read((char\*)this,sizeof(Item));**  **cout<<Ino<<”==>”<<Item<<endl;**  **File.close();**  **} void Item::Modify(int RecNo)**  **{**  **fstream File;**  **File.open(“STOCK.DAT”,ios::binary|ios::in|ios::out);**  **cout>>Ino;cin.getline(Item,20);**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //Statement 2**  **File.write((char\*)this,sizeof(Item));**  **File.close();**  **}** | 1 |
| b- | **Write a function in C++ to count the number of lines present in a text file**  **"STORY.TXT".** | 2 |
| c- | **Write a function in C++ to search for a BookNo from a binary file "BOOK.DAT", assuming the binary file is containing the objects of the following class.**  class  {  int Bno;  char Title[20];  public:  int RBno(){return Bno;}  void Enter(){cin>>Bno;gets(Title);}  void Display(){cout<<Bno<<Title<<endl;}  }; | 3 |
| Q5-  a- | **What do you understand by Degree and Cardinality of a table?** | 2 |
| b- | **Consider the following tables GAMES and PLAYER. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii) 6**  **Table: GAMES**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **GCode** | **GameName** | **Number** | **PrizeMoney** | **ScheduleDate** | | **101** | **Carom Board** | **2** | **5000** | **23-Jan-2004** | | **102** | **Badminton** | **2** | **12000** | **12-Dec-2003** | | **103** | **Table Tennis** | **4** | **8000** | **14-Feb-2004** | | **105** | **Chess** | **2** | **9000** | **01-Jan-2004** | | **108** | **Lawn Tennis** | **4** | **25000** | **19-Mar-2004** |     **Table: PLAYER**   |  |  |  | | --- | --- | --- | | **PCode** | **Name** | **Gcode** | | **1** | **Nabi Ahmad** | **101** | | **2** | **Ravi Sahai** | **108** | | **3** | **Jatin** | **101** | | **4** | **Nazneen** | **103** |   **(i) To display the name of all Games with their Gcodes**  **(ii) To display details of those games which are having PrizeMoney more than 7000.**    **(iii) To display the content of the GAMES table in ascending order of ScheduleDate.**  **(iv) To display sum of PrizeMoney for each of the Number of participation groupings (as shown in column Number 2 or 4)**   1. **SELECT COUNT(DISTINCT Number) FROM GAMES;**     **(vi)SELECT MAX(ScheduleDate),MIN(ScheduleDate) FROM GAMES;**    **(vii) SELECT SUM(PrizeMoney) FROM GAMES;**    **(viii) SELECT DISTINCT Gcode FROM PLAYER;** | 6 |
| Q6-  a- | **State and algebraically verify Absorption Laws.** | 2 |
| b- | Write the equivalent Boolean Expression for the following Logic Circuit | 2 |
| c- | **Write the SOP form of a Boolean function G, which is represented in a truth table as follows:**   |  |  |  |  | | --- | --- | --- | --- | | **P** | **Q** | **R** | **G** | | **0** | **0** | **0** | **0** | | **0** | **0** | **1** | **0** | | **0** | **1** | **0** | **1** | | **0** | **1** | **1** | **0** | | **1** | **0** | **0** | **1** | | **1** | **0** | **1** | **0** | | **1** | **1** | **0** | **1** | | **1** | **1** | **1** | **1** | | 1 |
| d- | 1. **Reduce the following Boolean Expression using K-Map: 3**   **F(A,B,C,D)=Σ(0,1,2,4,5,6,8,10)** | 3 |
| Q7-  a | **Define the term Bandwidth. Give unit of Bandwidth.** | 1 |
| b- | **Expand the following terminologies:**  **(i) CDMA (ii) GSM** | 1 |
| c- | **Define the term firewall.** | 1 |
| d- | What do you mean by Spam Mails? How can you protect your mailbox from Spams? | 1 |
| e | **Knowledge Supplement Organization has set up its new center at Mangalore for its office and web based activities. It has 4 blocks of buildings as shown in the diagram below:**  **Center to center distances between various block**s  Black A to Block B 50 m  Block B to Block C 150 m  Block C to Block D 25 m  Block A to Block D 170 m  Block B to Block D 125 m  Block A to Block C 90 m  **Number of Computers**  Block A 25  Block B 50  Block C 125  Block D 10  e1) Suggest a cable layout of connections between the blocks.  e2) Suggest the most suitable place (i.e. block) to house the server of this organization with a suitable reason.  e3) Suggest the placement of the following devices with justification  (i) Repeater  (ii) Hub/Switch  e4) The organization is planning to link its front office situated in the city in a hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed? | 4 |
| f- | **What is the importance of URL in networking?** | 1 |

Marking Scheme

Ans 1-a-

|  |  |
| --- | --- |
| **Actual Parameter** | **Formal Parameter** |
| It is a parameter, which is used in function call to send the value from calling environment | It is a parameter, which is used in function header, to receive the value from actual parameter |

(1 mark for each correct definition)

b-

i) math.h (ii) ctype.h

***(1/2 Mark for mentioning name of each header file)***

***c-***

include <iostream.h>

class PAYITNOW

{ int Charge;

public:

void Raise(){cin>>Charge;}

void Show(){cout<<Charge;} };

void main()

{ PAYITNOW P;

P.Raise();

P.Show(); }

*(1/2 Mark for correcting each error)*

OR

*(1 Mark for identifying all the 4 errors with no correction)*

*d-*

111:60

112:70

113:85

*(1 Mark for each correct line of output)*

*e-*

*@*a@E@E#rTH

***(1 Mark for writing all alphabets at correct positions)***

***(1/2 Mark for writing @ at correct positions)***

***(1/2 Mark for writing # at correct position)***

**f-**

(i) ABBC 2

*(2 Marks for mentioning correct option)*

*Q2-*

1. Data Encapsulation: Wrapping up of data and functions together in a single unit is known as Data Encapsulation. In a class, we wrap up the data and functions together in a single unit.

Data Hiding: Keeping the data in private visibility mode of the class to prevent it from

accidental change is known as Data Hiding.

class Computer

Data Hiding

{

char CPU[10]; int RAM;

public: Data Encapsulation

void STOCK();

void SHOW();

};

*( ½ Mark each for appropriate definitions)*

*(1 Mark for appropriate example showing both)*

*c-*

**class TEST**

**{ int TestCode;**

**char Description[20];**

**int NoCandidate,CenterReqd;**

**void CALCNTR();**

**public:**

**void SCHEDULE();**

**void DISPTEST();**

**};**

**void TEST::CALCNTR()**

**{ CenterReqd=NoCandidate/100 + 1;**

**}**

**void TEST::SCHEDULE()**

**{**

**cout<<”Test Code :”;cin>>TestCode;**

**cout<<”Description :”;gets(Description);**

**cout<<”Number :”;cin>>NoCandidate;**

**CALCNTR();**

**}**

**void TEST::DISPTEST()**

**{**

**cout<<”Test Code :”<<TestCode<<endl;**

**cout<<”Description :”<<Description<<endl;**

**cout<<”Number :”<<NoCandidate<<endl;;**

**cout<<”Centres :”<<CenterReqd<<endl;;**

**}**

***(1 Mark for correctly declaring Data Members)***

***(1 Mark for correctly defining CALCNTR())***

***( ½ Mark for correctly defining SCHEDULE())***

***( ½ Mark for calling CALCNTR() from SCHEDULE())***

***( ½ Mark for correctly defining DISPTEST())***

***( ½ Mark for correct syntax of class)***

***d-***

1. None of data members are accessible from objects belonging to class CUSTOMER.
2. Enter(), Show()
3. Data members: Voucher\_No, Sales\_Date, Salary

Member function: Sales\_Entry(), Sales\_Details(), Enter(), Show(), Register(), Status()

(iv) 66

***( 1 Mark for each correct answer)***

**Note:**

**No marks to be given for partial answers**

**Q3-**

**a-**

void AddNSave(int A[],int B[],int C[],int N)

{

for (int i=0;i<N;i++)

C[i]=2\*A[i]+3\*B[i];

}

***(1 Mark for function header with desired parameters)***

***(1 Mark for correct formation of loop)***

***(1 Mark for the formula)***

***(1 Mark for transferring elements in the resultant array)***

***b-***

Given,

W=2

N=40

M=30

Base(S)=5000

Row Major Formula:

Loc(S[I][J]) =Base(S)+W\*(M\*I+J)

Loc(S[20][10]) =5000+2\*(30\*20+10)

=5000+2\*(600+10)

=5000+1220

=6220

*(1 Mark for writing correct formula (for column major) OR substituting formula with*

*correct values)*

*(1 Mark for writing calculation step - at least one step)*

*(1 Mark for correct address)*

*c-*

**struct NODE**

**{**

**char Name[20];**

**NODE \*Link;**

**};**

**class QUEUE**

**{**

**NODE \*R,\*F;**

**public:**

**QUEUE();**

**void Insert();**

**void Delete();**

**};**

**void QUEUE::Insert()**

**{**

**NODE \*Temp;**

**Temp=new NODE;**

**gets(Temp->Name);**

**Temp->Link=NULL;**

**if (Rear==NULL)**

**{**

**Rear=Temp;**

**Front=Temp;**

**}**

**else**

**{**

**Rear->Link=Temp;**

**Rear=Temp;**

**}**

**}**

***( ½ Mark for appropriate function header)***

***( ½ Mark for declaring a Temporary pointer - TEMP)***

***(1 Mark for correct use of input/assignment of Temporary pointer- TEMP)***

***(1 Mark for checking FRONT as NULL and assigning REAR and FRONT as – TEMP) (1 Mark for connecting TEMP to link part of REAR and assigning REAR as TEMP)***

***d-***

void DiagSum(int M[][4],int N,int M)

{

int SumD1=0,SumD2=0;

for (int I=0;I<N;I++)

{

SumD1+=M[I][I];SumD2+=M[N-I-1][I];

}

cout<<"Sum of Diagonal 1:"<<SumD1<<endl;

cout<<"Sum of Diagonal 2:"<<SumD2<<endl;

22

**No. Answers Marks**

}

*( ½ Mark for correct function header)*

*( ½ Mark for initialization of SumD1 and SumD2 as 0)*

*( ½ Mark for appropriate loop)*

*( ½ Mark for correct expression for adding each diagonal elements)*

*e-* **Step 1: Push**

|  |
| --- |
|  |
|  |
|  |
| **True** |

**Step 2: Push**

|  |
| --- |
|  |
|  |
| **False** |
| **True** |

**Step 3: AND Push**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Pop** |  | **Pop** |  |
|  | **Op2=True** |  | **Op1=False** |  |
|  |  |  | **Op2=True** |  |
| **True** |  |  |  | **False** |

**Step 4: Push**

|  |
| --- |
|  |
|  |
| **True** |
| **False** |

**Step 5: Push**

|  |
| --- |
|  |
| **True** |
| **True** |
| **False** |

**Step 6: NOT Push**

|  |  |  |
| --- | --- | --- |
|  | **Pop** |  |
|  | **Op2=True** | **False** |
| **True** |  | **True** |
| **False** |  | **False** |

**Step 7: OR Push**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Pop** |  | **Pop** |  |
|  | **Op2=False** |  | **Op1=True** |  |
| **True** |  |  | **Op2=False** | **True** |
| **False** |  | **False** |  | **False** |

**Step 8: AND Push**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Pop** |  | **Pop** |  |
|  | **Op2=True** |  | **Op1=False** |  |
|  |  |  | **Op2=True** |  |
| **False** |  |  |  | **False** |

**Step 9: Pop**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  | **Result** |
|  | **False** |

***( 1½ Mark for showing stack position for operations NOT,OR and AND)***

***( ½ Mark for correctly evaluating the final result)***

***Q4-***

***a-***

**File.seekg(RecNo\*sizeof(Item));** //Statement 1

**File.seekp(RecNo\*sizeof(Item));** //Statement 2

***( ½ Mark for each correct statement)***

**b-**

void CountLine()

{

ifstream FIL("STORY.TXT");

int LINES=0;

char STR[80];

while (FIL.getline(STR,80))

LINES++;

cout<<"No. of Lines:"<<LINES<<endl;

f.close();

}

*(½ Mark for opening STORY.TXT correctly)*

*(½ Mark for initializing a counter variable as 0)*

*(½ Mark for correctly reading a line from the file)*

*(½ Mark for correctly incrementing the counter)*

c-

void BookSearch()

{

fstream FIL;

FIL.open("BOOK.DAT",ios::binary|ios::in);

BOOK B;

int bn,Found=0;

cout<<"Enter Book No. to search…"; cin>>bn;

while (FIL.read((char\*)&S,sizeof(S)))

if (FIL.RBno()==bn)

{

S.Display();

Found++;

}

if (Found==0) cout<<"Sorry! Book not found!!!"<<endl;

FIL.close();

}

*( ½ Mark for opening BOOK.DAT correctly)*

*( ½ Mark for reading each record from BOOK.DAT)*

*( ½ Mark for correct loop / checking end of file)*

*( 1 Mark for comparing Book number)*

*( ½ Mark for displaying the matching record)*

**ANs 5-**

**a-**

Degree: Number of Columns in a table

Cardinality: Number of rows in a table

*(1 Mark for each definition)*

**b-**

1. **display the name of all Games with their Gcodes**

Answer:

**SELECT GameName,Gcode FROM GAMES;**

***(1 mark for correct SELECTion of columns)***

**(ii) To display details of those games which are having PrizeMoney more than 7000.**

Answer:

# SELECT \* FROM GAMES WHERE PrizeMoney>7000

***( ½ mark for correct SELECTion of columns)***

***( ½ mark for correct use of WHERE)***

**(iii) To display the content of the GAMES table in ascending order of ScheduleDate.**

Answer:

**SELECT \* FROM GAMES ORDER BY ScheduleDate;**

***( ½ mark for correct SELECTion of columns)***

***( ½ mark for correct use of ORDER BY)***

1. **To display sum of PrizeMoney for each of the Number of participation groupings (as shown in column Number 2 or 4)**

Answer:

**SELECT SUM(PrizeMoney),Number FROM GAMES GROUP BY Number;**

***( ½ mark for correct SELECTion of columns)***

***( ½ mark for correct use of GROUP BY)***

1. **SELECT COUNT(DISTINCT Number) FROM GAMES;**

Answer:

**2**

***( ½ mark for correct output)***

**(vi)SELECT MAX(ScheduleDate),MIN(ScheduleDate) FROM GAMES;**

Answer:

**19-Mar-2004 12-Dec-2003**

***( ½ mark for correct output)***

**(vii) SELECT SUM(PrizeMoney) FROM GAMES;**

Answer:

**59000**

***( ½ mark for correct output)***

**(viii) SELECT DISTINCT Gcode FROM PLAYER;**

Answer:

**101**

**103**

**108**

***( ½ mark for correct output)***

***Ans 6 a-***

X+X.Y = X

L.H.S = X+X.Y

= X.1+X.Y

= X.(1+Y)

= X.1

= X

= R.H.S

X+X’.Y = X+Y

L.H.S. = X+X’.Y

= (X+X’).(X+Y)

= 1.(X+Y)

= X+Y

= R.H.S

***(1 mark for stating the correct law)***

***(1 mark for the appropriate verification using algebraic method)***

**b-**

*F=(P`+Q).(P+Q`)*

*(1 Mark for the correct expression )*

*c-*

(P,Q,R) = P’.Q.R’+P.Q’.R’+P.Q.R’+P.Q.R

**d-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **A’B’** | **A’B** | **AB** | **AB’** |
| **C’D’** | **1**  **0** | **1**  **4** | **12** | **1**  **8** |
| **C’D** | **1**  **1** | **1**  **5** | **13** | **9** |
| **CD** | **3** | **7** | **15** | **11** |
| **CD’** | **1**  **2** | **1**  **6** | **14** | **1**  **10** |

**F(A,B,C,D)=A’C’+A’D’+B’D’**

***(1 mark for correctly drawing K-Map with 1s represented on right places)***

***(1 mark for minimizing each Quad)***

***(1 mark for writing the complete Boolean Expression)***

***Ans 7-***

***a-***

Bandwidth is the capability of a medium to transmit an amount of information over a distance. Bandwidth of a medium is generally measured in bits per second (bps) or more commonly in kilobits per second (kbps)

***(½ Mark for correct definition and ½ Mark for correct unit.***

***b-***

1. Code Division Multiple Access
2. Global System for Mobile Communication

***(½ mark each expansion)***

***c-***

Firewall is a feature used for Network Security. In a Network there is always danger of information leaking out or leaking in. Firewall is a feature which forces all information entering or leaving the network to pass through a check to make sure that there is no unauthorized usage of the network.

***(1 Mark for correct definition)***

d- Spam mails, also known as junk e-mail, is a subset of spam that involves nearly 1

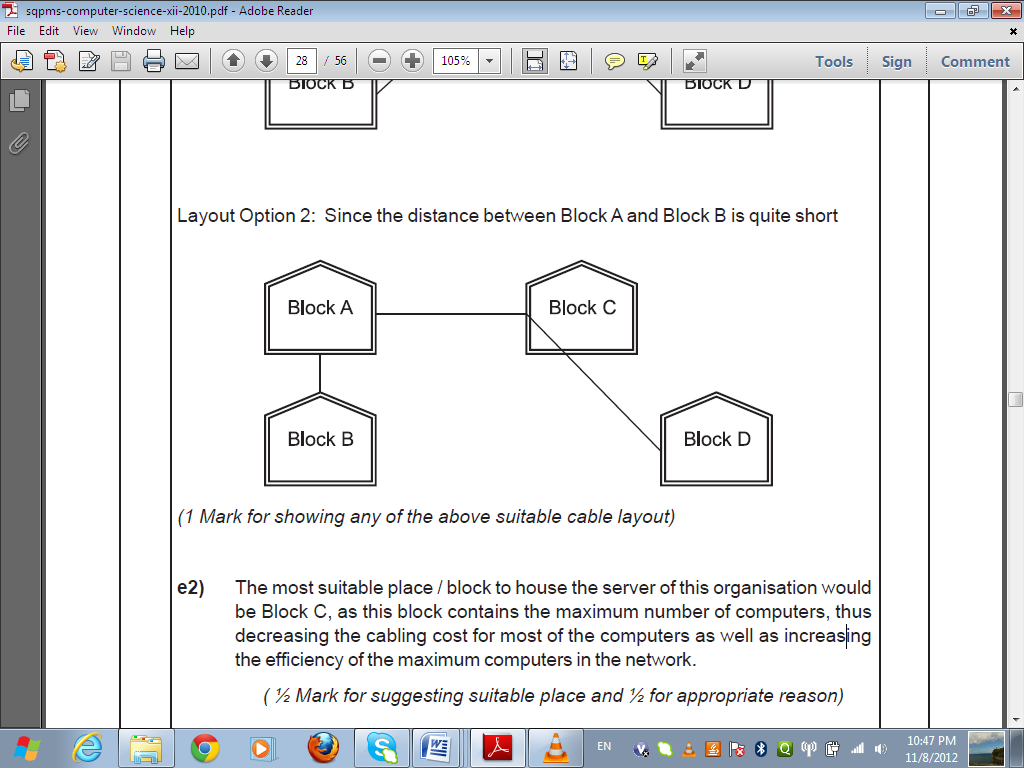
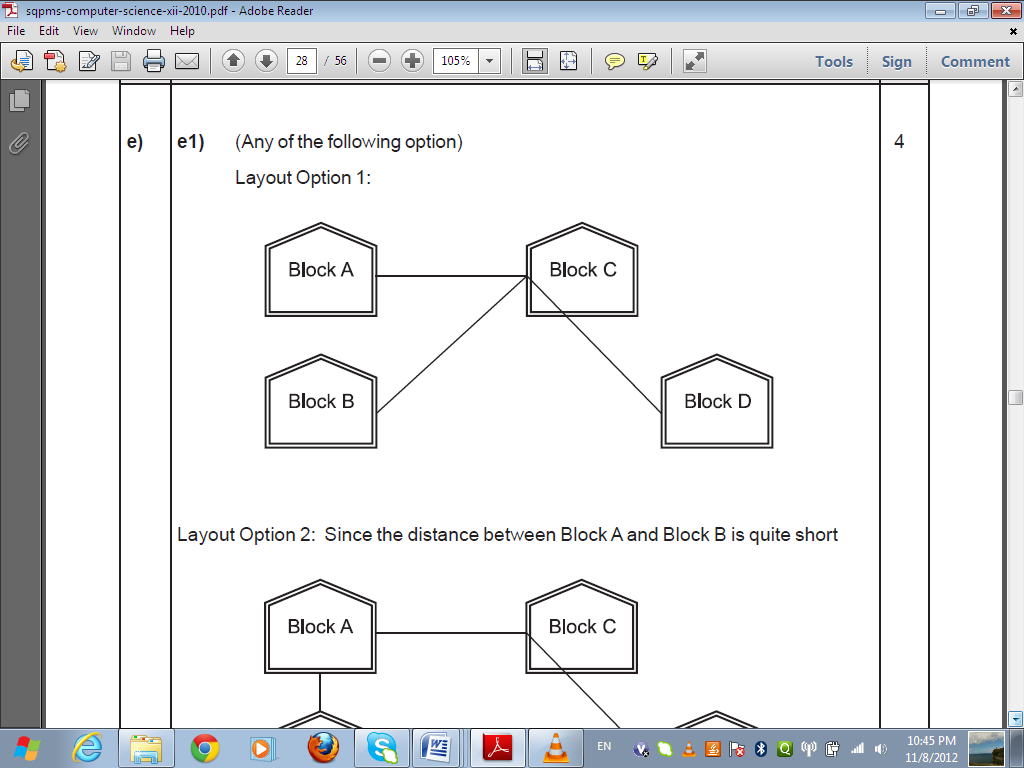
identical messages sent to numerous recipients by e-mail.

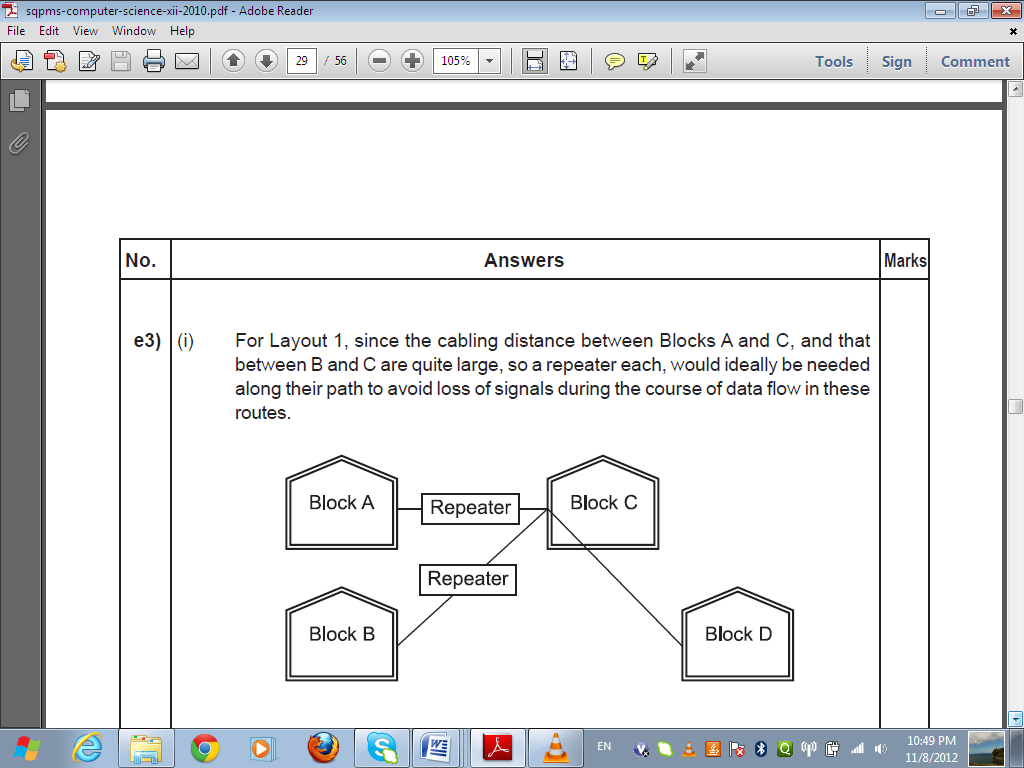
We can protect our mailbox from spams by creating appropriate filters.

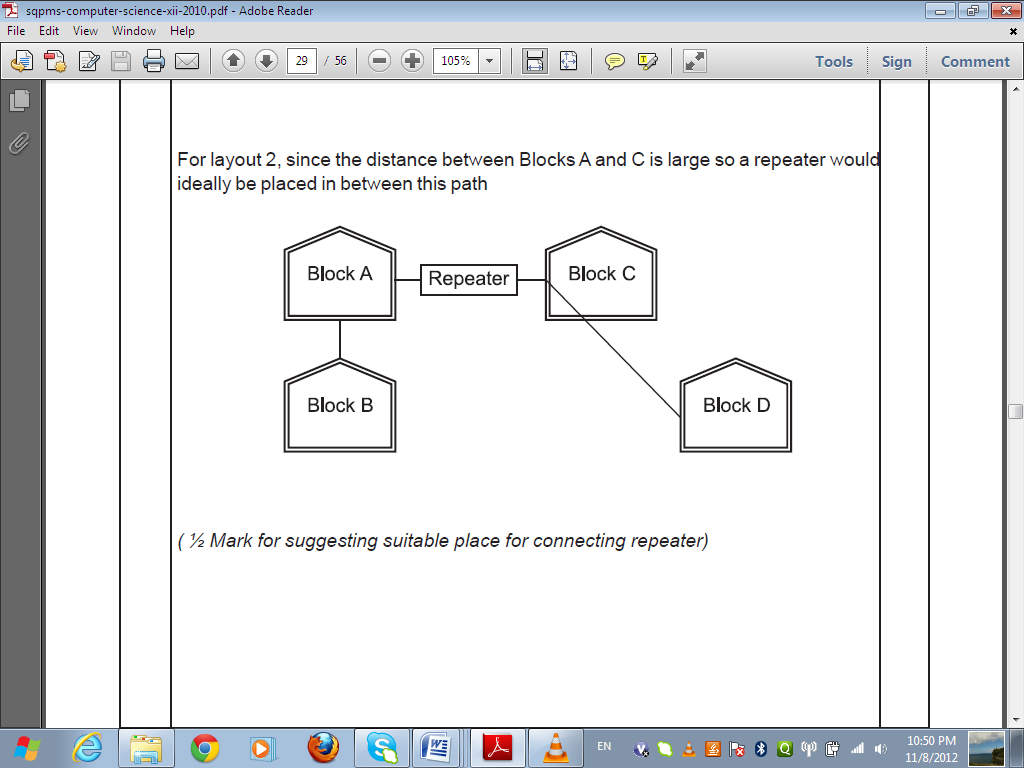
*( ½ Mark for the definition of Spam Mails)*

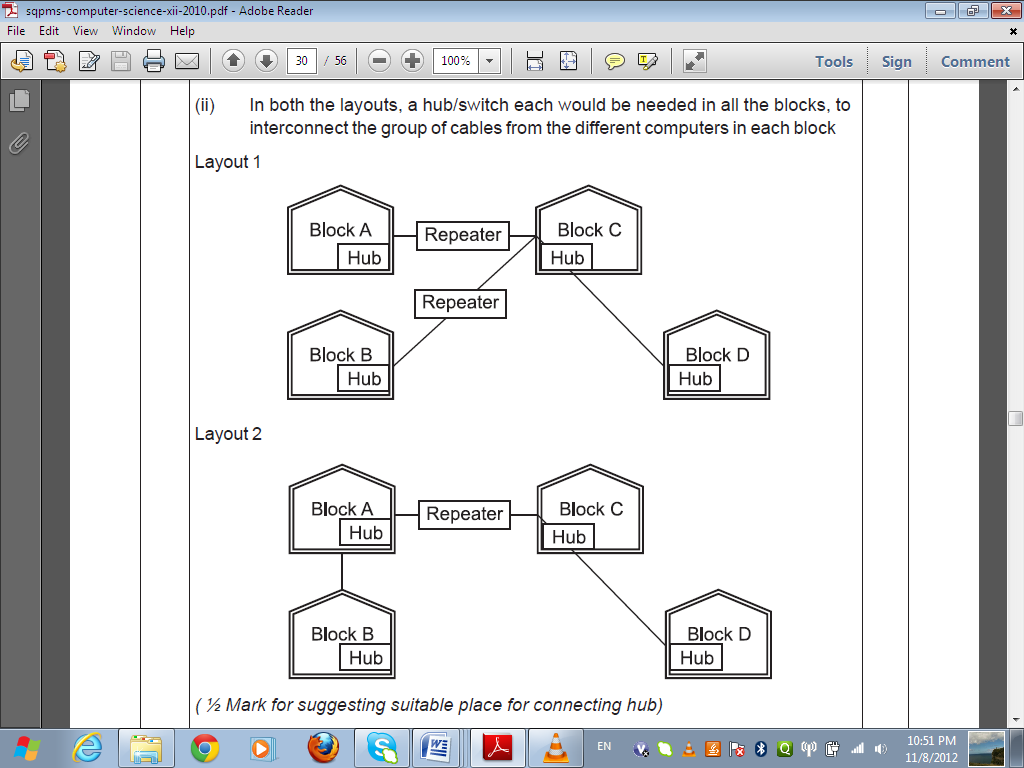
*( ½ Mark for the appropriate suggestion for protecting mailbox from it)*

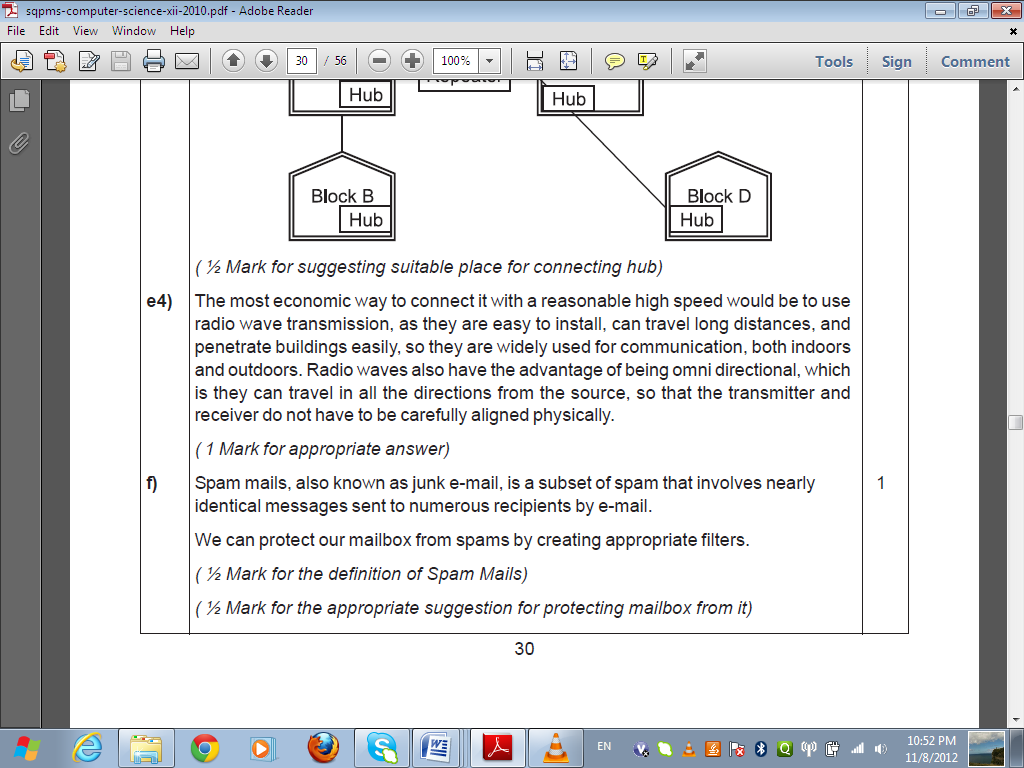
***e- e1-***

******





**

**

*f-*

URL stands for Uniform Resource Locator. Each page that is created for Web browsing is assigned a URL that effectively serves as the page’s worldwide name or address. URL’s have three parts: the protocol , the DNS name of the machine on which the page is located and a local name uniquely indicating the specific page(generally the filename).

***(1 Mark for correct significance)***