

**LAB MANUAL**

**SUBJECT: PS01CINT26-Advanced Programming Concepts & Data Structures**

**Academic year: 2018-19(ODD)  
  
 Semester: 1**

**PREPARED BY:  
 AVANISH MISHRA (18IT02)**

**Department of M.Sc. Information technology**

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **1.** | **Write a program to print hello world without using semicolon.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  if(cout<<"HELLO WORLD")  getch();  }  **OUTPUT:**  HELLO WORLD | |
| |  |  | | --- | --- | | **2.** | **Write a program to check whether a number is Armstrong or not.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int n,result;  cout<<"enter the value of n\n";  cin>>n;  int m=n;  result=0;  while(n!=0)  {  int rem=n%10;  result=result+rem\*rem\*rem;  n=n/10;  }  if(result==m)  {  cout<<"No. is armstrong\n";  }  else  cout<<"No. is not armstrong\n";  getch();  }  **OUTPUT:**  Enter the value of n:153  No. is Armstrong. | |
| |  |  | | --- | --- | | **3.** | **Write a program to check number is even or odd.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int a,b;  cout<<"Enter any number:"<<endl;  cin>>a;  if(a%2==0)  {  cout<<"NUMBER IS EVEN:";  }  else  {  cout<<"NUMBER IS ODD:";  }  getch();  }  **OUTPUT:**  Enter any number:85  NUMBER IS ODD: | |
| |  |  | | --- | --- | | **4.** | **Write a program to find area of a circle.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int r,area,pi=3.142;  cout<<"ENTER RADIUS OF CIRCLE:"<<endl;  cin>>r;  area=2\*pi\*r\*r;  cout<<"AREA OF CIRCLE IS:"<<area;  getch();  }  **OUTPUT:**  ENTER RADIUS OF CIRCLE:  2  AREA OF CIRCLE IS:25.136 | |
| |  |  | | --- | --- | | **5.** | **Find the number is Leap year or Not.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int a,b;  cout<<"ENTER THE YEAR WHICH YOU WANT TO CHECK:\n";  cin>>a;  if(a%4==0 && a%100!=0)  {  cout<<"YEAR IS A LEAP YEAR";  }  else  cout<<"YEAR IS NOT A LEAP YEAR";  getch();  }  **OUTPUT:**  ENTER THE YEAR WHICH YOU WANT TO CHECK:  2018  YEAR IS NOT A LEAP YEAR | |
| |  |  | | --- | --- | | **6.** | **Write a program to calculate simple interest.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int p,r,t,simpleinterest;  cout<<"Enter Principle amount:"<<endl;  cin>>p;  cout<<"Enter Rate:"<<endl;  cin>>r;  cout<<"Enter time:"<<endl;  cin>>t;  simpleinterest=(p\*r\*t)/100;  cout<<"SIMPLE INTEREST="<<simpleinterest;  getch();  }  **OUTPUT:**  Enter Principle amount:  7000  Enter Rate:  5  Enter time:  4  SIMPLE INTEREST=89 | |
| |  |  | | --- | --- | | **7.** | **Write a program to find the Fibonacci series of a number.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  int i=0,j=1,k,n,n1;  clrscr();  cout<<"Enter the value of n:"<<endl;  cin>>n;  cout<<"FIBONACCI SERIES="<<endl;  cout<<i<<endl<<j<<endl;  for(n1=0;n1<n-2;n1++)  {  k=i+j;  cout<<k<<endl;  i=j;  j=k;  }  getch();  }  **OUTPUT:**  Enter the value of n:  4  FIBONACCI SERIES=0 1 1 2 3 5 8 | |
| |  |  | | --- | --- | | **8.** | **Write a program to find the factorial of a number. (using recursion)** |   **SOLUTUON:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int n;  long int fact(int);  cout<<"Enter the number whose factorial you want to find:"<<endl;  cin>>n;  cout<<"The factorial of"<<" "<<n<<"is"<<fact(n)<<endl;  getch();  }  long fact(int num)  {  if(num==0)  return 1;  else  return num\*fact(num-1);  }  **OUTPUT:**  Enter the number whose factorial you want to find:  4  The factorial of 4 is 24 | |
| |  |  | | --- | --- | | **9.** | **An electricity board charges the following rates to domestic users to discourage large consumption of energy:**  **For the first 100 units:60P per unit.**  **For next 200 units:80P per unit.**  **Beyond 300 units :90P per unit.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  float unit,rate,amount;  cout<<"ENTER CONSUMED AMOUNT:";  cin>>unit;  if(unit>=0 && unit<=100)  {  amount=0.60\*unit+150;  cout<<"Your charge per unit is 0.60 paise"<<endl<<"Your billing amount is:"<<amount<<endl;  }  else if(unit>=101 && unit<=300)  {  amount=0.80\*unit+150;  cout<<"Your charge per unit is 0.80 paise"<<endl;  }  else if(unit>300)  {  amount=unit\*0.90+150;  cout<<"Your charge per unit is 0.90 paise"<<endl;  }  else  {  cout<<"Enter proper units!"<<endl;  }  if(amount>250)  {  amount=amount+(amount\*0.15);  cout<<"YOU HAVE EXCEEDED THE BILLING AMOUNT SO 0.15% EXTRA HAS BEEN CHARGED"<<endl;  cout<<"Your Bill is:"<<amount;  }  getch();  } **OUTPUT:** ENTER CONSUMED AMOUNT:200 Your charge per unit is 0.80 paise YOU HAVE EXCEEDED THE BILLING AMOUNT SO 0.15% EXTRA HAS BEEN CHARGED Your Bill is:356.5 | |
| |  |  | | --- | --- | | **10.** | **A company has following scheme for payment to their staff. Net salary = Gross salary – Deduction . Gross salary = Basic + DA + HRA + Medical. Deduction = Insurance + PF. Where DA(Dearness Allounce) = 40% of basic , HRA (house rent Allounce) = 15% of basic , medical = 5% of basic , PF(provident fund) = 7% of gross , Insurance =10% of gross.**  **Write a program to calculate net payment to any employee.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  float basic,da,hra,medical,pf,ins,ns,gs,ded;  cout<<"Enter Basic Salary of an employee:"<<endl;  cin>>basic;  da=basic\*0.4;  hra=basic\*0.15;  medical=basic\*0.05;  gs=basic+da+hra+medical;  cout<<"Gross Salary:"<<gs<<endl;  pf=gs\*0.07;  ins=gs\*0.1;  ded=ins+pf;  cout<<"Total Deduction:"<<ded<<endl;  ns=gs-ded;  cout<<"Net Salary:"<<ns<<endl;  getch();  }  **OUTPUT:** Enter Basic Salary of an employee: 85000 Gross Salary:136000 Total Deduction:23120 Net Salary:112880 | |
| |  |  | | --- | --- | | **11.** | **Write a program to swap of two number without using third variable.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int a,b;  cout<<"Enter value of a"<<endl;  cin>>a;  cout<<"Enter value of b"<<endl;  cin>>b;  cout<<"----BEFORE SWAPPING----"<<endl;  cout<<"a="<<a<<endl<<"b="<<b<<endl<<endl;  a=a+b;  b=a-b;  a=a-b;  cout<<"----AFTER SWAPPING----"<<endl<<endl;  cout<<"a="<<a<<endl<<"b="<<b;  getch();  } OUTPUT:  Enter value for a:  98  Enter value for b:  85  ----BEFORE SWAPPING----  a=98  b=85  ----AFTER SWAPPING----  a=85  b=98 | |
| |  |  | | --- | --- | | **12.** | **Write a program to read four numbers from keyboard and find out maximum out of these four (nested if else).** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  int a,b,c,d,m;  clrscr();  cout<<"Enter 4 values:\n";  cin>>a>>b>>c>>d;  if(a>b)  {  if(a>c)  m=a;  else  m=c;  }  else  {  if(b>c)  m=b;  else  m=c;  }  if(m>d)  {  cout<<"Greater value is:"<<m;  }  else  {  cout<<"Greater value is:"<<d;  }  getch();  }  **OUTPUT:**  Enter 4 values:  85 95 97 99  Greater value is:99 | |
| |  |  | | --- | --- | | **13.** | **Write a program to read three numbers from keyboard and find out maximum out of these ternary.(using ternary operator)** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  int a,b,c,max;  clrscr();  cout<<"Enter 3 values:\n";  cin>>a>>b>>c;  max=((a>b)&&(a>c))?a:(b>c)?b:c;  cout<<"Maximum value:"<<max;  getch();  }  **OUTPUT:**  Enter three numbers:  23 85 99  Maximum value is:99 | |
| |  |  | | --- | --- | | **14.** | **Write a program to read marks from keyboard and your program should display equivalent grade according to following table.(if-else if ladder)**  **Marks Grade**  **100-80 Honours.**  **60-79 First class.**  **40-59 Second class.**  **0-39 Fail.** |   **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int a;  cout<<"Enter the obtained marks:";  cin>>a;  if(a<=100 && a>=80)  {  cout<<"STUDENT HAS ACHIEVED HONOURS!";  }  else if(a<=79 && a>=60)  {  cout<<"STUDENT HAS ACHIEVED FIRST CLASS!";  }  else if(a<=59 && a>=40)  {  cout<<"STUDENT HAS ACHIEVED SECOND CLASS!";  }  else if(a<=39 && a>=0)  {  cout<<"FAILED!\nPRACTICE MAKES MAN PERFECT!!";  }  else  {  cout<<"ENTER APPROTIATE MARKS!";  }  getch();  } **OUTPUT:**  Enter the obtained marks:  80  STUDENT HAS ACHIEVED SECOND CLASS! | |
| |  |  | | --- | --- | | **15.** | **Write a program that reads a number from 1 to 7 and accordingly it should display Monday to Sunday(switch-case).** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int a;  cout<<"ENTER THE NUMBER OF DAY:";  cin>>a;  switch(a)  {  case 1:  cout<<"THIS IS SUNDAY!";  break;  case 2:  cout<<"THIS IS MONDAY!";  break;  case 3:  cout<<"THIS IS TUESDAY!";  break;  case 4:  cout<<"THIS IS WEDNESDAY!";  break;  case 5:  cout<<"THIS IS THURSDAY!";  break;  case 6:  cout<<"THIS IS FRIDAY!";  break;  case 7:  cout<<"THIS IS SATURDAY";  break;  default:  cout<<"ENTER VALID NUMBER!!";  }  getch();  } **OUTPUT:** ENTER THE NUMBER OF DAY: 2 THIS IS MONDAY! | |
| |  |  | | --- | --- | | **16.** | **Write a program to check whether input alphabet is a vowel or not.** |   **SOLUTION:-**  #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  char p;  cout<<"ENTER AN ALPHABET:\n";  cin>>p;  if(p=='a'|| p=='A' || p=='e' || p=='E' || p=='i' || p=='I' || p=='o' || p=='O' || p=='u' || p=='U')  cout<<p<<"IS A VOWEL!";  else  cout<<p<<" IS NOT A VOWEL!";  getch();  } **OUTPUT:** ENTER AN ALPHABET:  m  m IS NOT A VOWEL! | |
| |  |  | | --- | --- | | **17.** | **Write a program to add n numbers .** |   **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int n,result=0,rem;  cout<<"Enter the count of number you want to add:";  cin>>n;  while(n!=0)  {  rem=n%10;  result=result+rem;  n=n/10;  }  cout<<"Addition is:"<<result;  getch();  } **OUTPUT:** Enter the number you want to add:  23  Addition is:5 | |
| |  |  | | --- | --- | | **18.** | **Write a program to reverse a numbers.** |   **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int n,rem,result;  cout<<"ENTER THE NUMBER:";  cin>>n;  cout<<"REVERSE OF THE NUMBER IS:";  while(n!=0)  {  rem=n%10;  n=n/10;  cout<<rem;  }  getch();  }  **OUTPUT:**  ENTER THE NUMBER:  851  REVERSE OF THE NUMBER IS:158 | |
| **19.** | **Write a program to palindrome of number.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int n,m,result=0,rem;  cout<<"Enter the value of N:";  cin>>n;  m=n;  while(n!=0)  {  rem=n%10;  result=result\*10+rem;  n=n/10;  }  if(result==m)  {  cout<<result<<" "<<"IS PALINDROME!";  }  else  {  cout<<"NOT A PALINDROME NUMBER!";  }  getch();  } OUTPUT:  Enter the value of N:121 121 IS PALINDROME! |
| **20.** | **Write a program for addition of two matrices.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int a[3][3],i,j,b[3][3],c[3][3];  cout<<"---------- FOR FIRST MATRIX ----------"<<endl;  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  int val;  cout<<"Enter the value for index"<<i<<j<<endl;  cin>>val;  a[i][j]=val;  }  }  cout<<"First matrix is"<<endl;  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  cout<<" "<<a[i][j];  }  cout<<endl;  }  cout<<"---------- FOR SECOND MATRIX ----------"<<endl;  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  int val;  cout<<"Enter value for index"<<i<<j<<endl;  cin>>val;  b[i][j]=val;  }  }  cout<<"First Matrix is:"<<endl;  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  cout<<" "<<b[i][j];  }  cout<<endl;  }  cout<<"---------- ADDITION OF MATRIX ----------"<<endl;  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  cout<<" "<<a[i][j];  }  cout<<endl;  }  cout<<"+"<<endl;  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  cout<<" "<<b[i][j];  }  cout<<endl;  }  cout<<"="<<endl;  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  c[i][j]=a[i][j]+b[i][j];  cout<<" "<<c[i][j];  }  cout<<endl;  }  getch();  } **OUTPUT:** |
| **21. A** | **Print the below patterns:**  **\*  \*\*\*  \*\*\*\*\*  \*\*\*\*\*\*\*  \*\*\*\*\*\*\*\*\*** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int i,j;  for(i=1;i<=9;i++)  {  if(i%2==0)  continue;  for(int k=9;k>=i;k--)  {  cout<<" ";  }  for(j=1;j<=i;j++)  {  cout<<" \*";  }  cout<<"\n";  }  getch();  } |
| **B.** | **12345  1234  123  12   1** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  int i,j,k;  clrscr();  for(i=5;i>=1;i--)  {  for(k=4;k>=1;k--)  {  cout<<" ";  }  for(j=1;j<=i;j++)  {  cout<<" "<<j;  }  cout<<endl;  }  getch();  } |
| **C.** | **\*\*\*\*\*\*\***  **\*\*\* \*\*\***  **\*\* \*\***  **\* \*** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  int i,j,k;  clrscr();  for(i=4;i>=0;i--)  {  for(j=0;j<i;j++)  {  cout<<"\*";  }  if(i<4)  {  for(k=4;k>i;k--)  {  cout<<" ";  }  }  for(j=0;j<i;j++)  {  cout<<"\*";  }  cout<<endl;  }  getch();  } |
| **D.** | **A B C D E F G H  A B C D E F G  A B C D E F  A B C D E  A B C D  A B C  A B   A** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  char x;  clrscr();  for(int i=8;i>=1;i--)  {  for(int k=8;k>i;k--)  {  cout<<" ";  }  for(int j=1;j<=i;j++)  {  x=j+64;  cout<<" "<<x;  }  cout<<endl<<endl;  }  getch();  } |
| **22.** | **Write a C++ program to use scope resolution operator. Display the various values of the same variables declared at different scope levels.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  char a='P';  void main()  {  clrscr();  char a='M';  cout<<"LOCAL VARIALBE IS:"<<a<<endl;  cout<<"GLOBAL VARIABLE IS:"<<::a;  getch();  } **OUTPUT:** LOCAL VARIABLE IS:P GLOBAL VARIABLE IS:M |
| **23.** | **a) Design a simple class with all arithmetic function. Use them in MAIN function.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Calc  {  public:  void add(int a,int b)  {  int c;  c=a+b;  cout<<"Addition is:"<<c<<endl;  }  void sub(int a,int b)  {  int c;  c=a-b;  cout<<"Subtraction is:"<<c<<endl;  }  void mul(int a,int b)  {  int c;  c=a\*b;  cout<<"Multiplication is:"<<c<<endl;  }  void div(int a,int b)  {  int c;  c=a/b;  cout<<"Division is:"<<c<<endl;  }  };  void main()  {  Calc obj;  clrscr();  int a,b;  cout<<"Enter the value of a:"<<endl;  cin>>a;  cout<<"Enter the value of b:"<<endl;  cin>>b;  obj.add(a,b);  obj.sub(a,b);  obj.mul(a,b);  obj.div(a,b);  getch();  } **OUTPUT:** Enter value of a:2 Enter value of b:1 Addition is:3 Subtraction is:1 Multiplication is:2 Division is:1 |
|  | **b) Create a class named Student having data member name, rollno and three marks. Write a member function to  (1) Input name, rollno and marks. (2) Calculate percentage.  (3) Display all the information.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Student  {  public:  char name;  int rno,m1,m2,m3,percentage;  void input()  {  cout<<"Enter RollNo:";  cin>>rno;  cout<<"Enter Subject1 Marks:"<<endl;  cin>>m1;  cout<<"Enter Subject2 Marks:"<<endl;  cin>>m2;  cout<<"Enter Subject3 Marks:"<<endl;  cin>>m3;  }  void calc()  {  percentage=(m1+m2+m3)/3;  }  void display()  {  cout<<"Marks for Subject1 is:"<<m1<<endl;  cout<<"Marks for Subject2 is:"<<m2<<endl;  cout<<"Marks for Subject3 is:"<<m3<<endl;  cout<<"Percentage is:"<<percentage<<endl;  }  };  void main()  {  clrscr();  Student s;  s.input();  s.calc();  s.display();  getch();  } **OUTPUT:** Enter RollNo. 359 Enter marks for Subject1:20 Enter marks for Subject2:20 Enter marks for Subject3:20 Marks for Subject1 is:20 Marks for Subject2 is:20 Marks for Subject3 is:20 Percentage is:20% |
|  | **c) Declare a class cricketer including following member. Data member:  1. Name  2. Age  3. For which country he plays  4. Type of cricketer (batsman/ bowler)  5. Total matches he has played.  Member function:  1. To assign initial values  2. To enter data  3. To display his data. Write a main program for 5 cricketer.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Cricketer  {  public:  char name[20],country[10],type[10];  int age,match;  Cricketer()  {  name[20]='/0';  country[10]='/0';  type[10]='/0';  age=0;  match=0;  }  void get()  {  cout<<"Enter players name:"<<endl;  cin>>name;  cout<<"Enter age:"<<endl;  cin>>age;  cout<<"Enter players country:"<<endl;  cin>>country;  cout<<"Enter type of cricketer:"<<endl;  cin>>type;  cout<<"Enter total number of matches played:";  cin>>match;  }  void display()  {  cout<<"Player's Name is:"<<name<<endl;  cout<<"Player's Age is:"<<age<<endl;  cout<<"Player's country is:"<<country<<endl;  cout<<"Player's type is:"<<type<<endl;  cout<<"Total number of match a player has played is:"<<match<<endl;  }  };  void main()  {  clrscr();  int i;  Cricketer C;  for(i=1;i<=5;i++)  {  cout<<"DETAILS OF THE PLAYER"<<i<<endl;  C.get();  }  for(i=1;i<=5;i++)  {  cout<<"-----DEATILS OF THE PLAYER-----"<<i<<endl;  C.display();  }  getch();  } **OUTPUT:**  C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (17).png |
|  | **d) Write a program to multiply two matrices A and B by using class.(Size of matrices are 4 \* 4)** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  void main()  {  clrscr();  int a[4][4],i,j,b[4][4],c[4][4];  cout<<"---------- FOR FIRST MATRIX ----------"<<endl;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  int val;  cout<<"Enter the value for index"<<i<<j<<endl;  cin>>val;  a[i][j]=val;  }  }  cout<<"First matrix is"<<endl;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  cout<<" "<<a[i][j];  }  cout<<endl;  }  cout<<"---------- FOR SECOND MATRIX ----------"<<endl;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  int val;  cout<<"Enter value for index"<<i<<j<<endl;  cin>>val;  b[i][j]=val;  }  }  cout<<"First Matrix is:"<<endl;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  cout<<" "<<b[i][j];  }  cout<<endl;  }  cout<<"---------- ADDITION OF MATRIX ----------"<<endl;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  cout<<" "<<a[i][j];  }  cout<<endl;  }  cout<<"+"<<endl;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  cout<<" "<<b[i][j];  }  cout<<endl;  }  cout<<"="<<endl;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  c[i][j]=0;  for(int k=0;k<4;k++)  {  c[i][j]=c[i][j]+a[i][k]\*b[k][j];  }  }  cout<<c[i][j]<<" ";  cout<<endl;  }  getch();  }  **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (19).png |
| **24.** | **a) Design classes named Triangle, Square, and Circle. Make the different function in each class to find areas of particular shape.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Circle  {  public:  float r,a;  void get()  {  cout<<"Enter radius of circle:"<<endl;  cin>>r;  }  void display()  {  a=3.14\*r\*r;  cout<<"AREA OF CIRCLE IS:"<<a<<endl;  }  };  class Square  {  public:  int side,sq;  void put()  {  cout<<"Enter side of square:"<<endl;  cin>>side;  }  void show()  {  sq=side\*side;  cout<<"AREA OF SQUARE IS:"<<sq<<endl;  }  };  class Triangle  {  public:  int b,h,tri;  void input()  {  cout<<"Enter base of triangle:"<<endl;  cin>>b;  cout<<"Enter height of triangle:"<<endl;  cin>>h;  }  void output()  {  tri=1/2\*b\*h;  cout<<"AREA OF TRIANGLE IS:"<<tri<<endl;  }  };  void main()  {  clrscr();  Circle c;  Square s;  Triangle t;  c.get();  c.display();  s.put();  s.show();  t.input();  t.output();  getch();  }  **OUTPUT:** Enter radius of circle:2 Area Of Circle is:12.56 Enter side of Square:2 Area Of Square is:4 Enter Base of Triangle:2 Enter Height of Triangle:4 Area of Triangle is:4 |
|  | **b) Define a class to represent a bank account. Include the members like name of the depositor, account number, type of account, and balance amount in the account. Make functions (1) To assign initial values, (2) To deposit an amount, (3) To withdraw an amount after checking the balance, (4) To display name and balance. Write a main program to test the program.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  #include<string.h>  #include<stdlib.h>  class Bank  {  public:  char name[20],type[20];  int accno,deposit,newbal,withdraw;  Bank()  {  name[20]='/0';  type[20]='/0';  accno=0;  deposit=0;  newbal=0;  }  void assign()  {  cout<<"Enter name of account holder"<<endl;  cin>>name;  cout<<"Enter type of account(Current/Savings):"<<endl;  cin>>type;  cout<<"Enter account number:"<<endl;  cin>>accno;  cout<<"Enter opening amount:"<<endl;  cin>>newbal;  while(newbal<500)  {  cout<<"Minimum amount is 500!"<<endl;  cout<<"Re-enter the amount:"<<endl;  cin>>newbal;  }  }  void data()  {  cout<<"Enter the amount to be deposited:"<<endl;  cin>> deposit;  newbal=deposit+newbal;  cout<<"AMOUNT DEPOSITED!"<<endl;  cout<<"Your balance is:"<<newbal<<endl;  }  void remove()  {  cout<<"Enter the amount to be withdrawn:"<<endl;  cin>>withdraw;  cout<<"======================"<<endl;  if(withdraw<500)  {  cout<<"NOT ENOUGH BALANCE!!"<<endl;  cout<<"TRANSACTION CANCELED!"<<endl;  }  else  {  newbal=newbal-withdraw;  cout<<"Withdrawl amount is:"<<withdraw<<endl;  cout<<"BALANCE WITHDRAWN SUCCESSFULLY!"<<endl;  }  }  void display()  {  cout<<"--------------------"<<endl;  cout<<"Account name is:"<<name<<endl;  cout<<"Account number is:"<<accno<<endl;  cout<<"Account type is:"<<type<<endl;  cout<<"Total balance is:"<<newbal<<endl;  cout<<"--------------------"<<endl;  }  };  void main()  {  Bank b;  int n=1;  int ch=1;  while(n!=0)  {  clrscr();  cout<<"1.NEW"<<endl;  cout<<"2.DEPOSIT"<<endl;  cout<<"3.WITHDRAWAL"<<endl;  cout<<"4.DISPLAY"<<endl;  cout<<"5.EXIT"<<endl;  cout<<"Enter your choice:"<<endl;  cin>>ch;  switch(ch)  {  case 1:b.assign();  break;  case 2:b.data();  break;  case 3:b.remove();  break;  case 4:b.display();  break;  case 5:exit(1);  break;  }  cout<<"Press Any Key to Continue....";  getch();  }  getch();  }  **OUTPUT:**  C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (20).png**C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (21).png** |
| **25.** | **a) Write a program to overload squareroot() function which finds square root of integer, long and double numbers.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  #include<math.h>  class Square  {  public:  void squareroot(int x)  {  int a;  cout<<"Enter int value"<<endl;  cin>>a;  }  void squareroot(long y,int q)  {  int b;  cout<<"Enter long value:"<<endl;  cin>>b;  }  void squareroot(double z,int w,int e)  {  int c;  cout<<"Enter double value:"<<endl;  cin>>c;  }  void display();  {  cout<<"=========="<<endl;  cout<<"INT VALUE IS:"<<a<<endl;  cout<<"LONG VALUE IS:"<<b<<endl;  cout<<"DOUBLE VALUE IS:"<<c<<endl;  cout<<"==========";  }  };  void main()  {  clrscr();  Square S;  S.squareroot(2);  S.squareroot(1000.99,3);  S.squareroot(12.33,4,5);  S.display();  getch();  }  **OUTPUT:** Enter INT Value:85 Enter LONG Value:22345678 Enter DOUBLE Value:98.234567891234567 INT VALUE IS:85  LONG VALUE IS:22345678  DOUBLE VALUE IS: 98.234567891234567 |
|  | **b) Create two classes DM and DB which store the value of distances. DM stores distances in meter and centimeters and DB in feet and inches. Write a program that read values of the class objects and add one object of DM with another object of DB.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class DB;  class DM  {  float m,c;  public:  void get()  {  cout<<"Enter meter:"<<endl;  cin>>m;  cout<<"Enter centimeter:"<<endl;  cin>>c;  }  void display()  {  cout<<"meter="<<m<<endl<<"centimeter="<<c<<endl;  }  friend void sum(DM,DB);  };  class DB  {  float f,i;  public:  void get()  {  cout<<"Enter feet:"<<endl;  cin>>f;  cout<<"Enter inch:"<<endl;  cin>>i;  f=f\*0.30;  i=i\*2.50;  }  void display()  {  cout<<"meter="<<f<<endl<<"centimeter="<<i<<endl;  }  friend void sum(DM,DB);  };  void sum(DM M,DB B)  {  int x=M.c+B.i;  float y=x/100;  x=x%100;  y=y+M.m+B.f;  cout<<endl;  cout<<"Sum of DM & DB values are:"<<endl<<y<<" meter"<<" and "<<x<<" centimeter.";  }  void main()  {  clrscr();  DM M;  DB B;  cout<<"Enter value for DM:"<<endl;  M.get();  cout<<"Enter value for DB:"<<endl;  B.get();  cout<<"Values for DM:"<<endl;  M.DM::display();  cout<<"Values for DB:"<<endl;  B.DB::display();  sum(M,B);  getch();  } **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (23).png |
|  | **c) Write a program to add two complex numbers using friend function.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class second;  class first  {  private:  int a,b;  public:  void get()  {  cout<<"\nEnter value of a:";  cin>>a;  cout<<"\nEnter value of b:";  cin>>b;  }  friend void sum(first f,second s);  void display()  {  cout<<endl;  cout<<a<<"+"<<b<<"i"<<endl;  }  };  class second  {  private:  int c,d;  public:  void get1()  {  cout<<"\nenter value of c:";  cin>>c;  cout<<"\nenter value of d:";  cin>>d;  }  friend void sum(first f,second s);  void display()  {  cout<<endl;  cout<<c<<"+"<<d<<"i"<<endl;  }  };  void sum(first f,second s)  {  int x=f.a+s.c;  int y=f.b+s.d;  cout<<endl;  cout<<x<<"+"<<y<<"i";  }  void main()  {  first f;  second s;  clrscr();  f.get();  s.get1();  f.display();  cout<<" +";  s.display();  cout<<" =";  sum(f,s);  getch();  } **OUTPUT: C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (24).png** |
| **26.** | **Write a program to perform operations like addition, multiplication and subtraction using operator overloading.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class First  {  public:  int a,b;  void get()  {  cout<<"\nEnter value of a:";  cin>>a;  }  void get1()  {  cout<<"\nEnter value of b:";  cin>>b;  }  int operator +(First s)  {  return(a+s.b);  }  int operator -(First s)  {  return(a-s.b);  }  int operator \*(First s)  {  return(a\*s.b);  }  };  void main()  {  First f,s;  clrscr();  f.get();  s.get1();  int res1,res2,res3;  res1=f+s;  res2=f-s;  res3=f\*s;  cout<<"a+b:"<<res1<<endl;  cout<<"a-b:"<<res2<<endl;  cout<<"a\*b:"<<res3<<endl;  getch();  } **OUTPUT:** Enter value of a:84 Enter value of b:1 a+b=85 a-b=83 a\*b=84 |
| 27. | **a) Design a class called Vehicle having a unique vehicle-number. Derive a class called Car having members as no. of seats and max. speed limit (in kms/hr.). Derive second class called Truck having members as the capacity of load (in tons) and max. speed limit (kms/hr.). Main should call a function, which will compare the Speed of a car and a truck and print the speedier one. Main should also print the vehicle no. of all the vehicles Whose speed limit is more than 150 kms/hr.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Vehicle  {  public:  int uno;  };  class Car:public Vehicle  {  public:  int seats,sp;  void get()  {  cout<<"Enter Vehicle Number:";  cin>>uno;  cout<<"\nEnter No of seats:";  cin>>seats;  cout<<"\nEnter Speed Limit(km/hr):";  cin>>sp;  }  };  class Truck:public Vehicle  {  public:  int load,sl;  void get1()  {  cout<<"Vehicle Number:";  cin>>uno;  cout<<"\nCapacity of Load(Tons):";  cin>>load;  cout<<"\nSpeed Limit(km/hr):";  cin>>sl;  }  };  void main()  {  Vehicle V;  Car C;  Truck T;  clrscr();  cout<<"Enter value for Car:"<<endl;  C.get();  cout<<"Enter value for Truck:"<<endl;  T.get1();  if(C.sp > T.sl)  {  cout<<"\nCar is faster";  }  else  {  cout<<"\nTruck is faster";  }  if(C.sp>150)  {  cout<<"\nUnique Vehicle Number:"<<C.uno;  }  if(T.sl>150)  {  cout<<"\nUnique Vehicle Number:"<<T.uno;  }  getch();  }  **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (3).png |
|  | **b) Create a class student whose members are name, college name. Create two derived classes for IT students and commerce students. Develop a virtual function that finds whether a particular student passed with distinctions or not. IT students get distinction if they secure marks 66% or above, for commerce student it is 70% or above. Display the name of students graduating with distinction.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Student  {  public:  char name[50],clg[50];  float per;  int total,m1,m2,m3;  virtual void show()=0;  };  class IT:public Student  {  public:  void get()  {  cout<<"\nEnter name:";  cin>>name;  cout<<"\nEnter college name:";  cin>>clg;  cout<<"\nEnter marks1:";  cin>>m1;  cout<<"\nEnter marks2:";  cin>>m2;  cout<<"\nEnter marks3:";  cin>>m3;  total=m1+m2+m3;  per=(total\*100)/300;  }  void show()  {  if(per>=66)  {  cout<<"\nName:"<<name;  cout<<"\nPercent:"<<per;  }  }  };  class Commerce:public Student  {  public:  void get1()  {  cout<<"\nEnter name:";  cin>>name;  cout<<"\nEnter college name:";  cin>>clg;  cout<<"\nEnter marks1:";  cin>>m1;  cout<<"\nEnter marks2:";  cin>>m2;  cout<<"\nEnter marks3:";  cin>>m3;  total=m1+m2+m3;  per=(total\*100)/300;  }  void show()  {  if(per>=70)  {  cout<<"\nName:"<<name;  cout<<"\nPercent:"<<per;  }  }  };  void main()  {  Student \*ptr;  IT I;  Commerce C;  clrscr();  cout<<"\nEnter values for IT student";  I.get();  cout<<"\nEnter values for Commerce student";  C.get1();  ptr=&I;  ptr->show();  ptr=&C;  ptr->show();  getch();  } **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (5).png |
| **28.** | **a) Declare a class called bird having data members as name and color. Derive two classes called flying and non-flying has data member speed\_to\_walk. Declare a pure virtual function get\_speed and display in base class bird. Override these two in the derived classes for getting speed and displaying all data members of the class.** |
|  | **SOLUTION:-** include<conio.h>  class Bird  {  public:  char name[50],colour[50];  virtual void get\_speed()=0;  virtual void display()=0;  };  class Flying:public Bird  {  public:  float speed\_to\_walk;  void get\_speed()  {  cout<<"\nEnter Name:";  cin>>name;  cout<<"\nEnter Colour:";  cin>>colour;  cout<<"\nEnter speed of walking:";  cin>>speed\_to\_walk;  }  void display()  {  cout<<"\nName is:"<<name;  cout<<"\nColour is:"<<colour;  cout<<"\nSpeed is:"<<speed\_to\_walk;  }  };  class Non\_flying:public Bird  {  public:  float speed\_to\_walk;  void get\_speed()  {  cout<<"\nEnter name:";  cin>>name;  cout<<"\nEnter colour:";  cin>>colour;  cout<<"\nEnter speed of walking:";  cin>>speed\_to\_walk;  }  void display()  {  cout<<"\nName:"<<name;  cout<<"\nColour:"<<colour;  cout<<"\nSpeed:"<<speed\_to\_walk;  }  };  void main()  {  Bird \*ptr;  Flying F;  Non\_flying N;  clrscr();  ptr=&F;  cout<<"\nEnter values for Flying Type:";  ptr->get\_speed();  ptr=&N;  cout<<"\nEnter values for Non\_flying Type:";  ptr->get\_speed();  ptr=&F;  cout<<"\nValues for Flying Type:";  ptr->display();  ptr=&N;  cout<<"\nValues for Non\_flying Type:";  ptr->display();  getch();  } **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (6).png |
|  | **b) Declare a class Person having name as member. Derive two classes 1. Businessman having income and number of people involved in his business as members. 2. Employee having income as a member. 3. Main should call a function that compares the incomes of the above two categories and print the larger one. Main should also find businessman having more than 10 people involved in their business.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Employee;  class Businessman;  class Person  {  public:  char name[50];  };  class Businessman:public Person  {  public:  float income;  int people;  void get();  friend void compare(Businessman B, Employee E);  };  class Employee:public Person  {  public:  float income;  void get();  friend void compare(Businessman B, Employee E);  };  void Businessman::get()  {  cout<<"\nEnter name:";  cin>>name;  cout<<"\nEnter income:";  cin>>income;  cout<<"\nEnter no of People:";  cin>>people;  }  void Employee::get()  {  cout<<"\nEnter name:";  cin>>name;  cout<<"\nEnter income:";  cin>>income;  }  void compare(Businessman B, Employee E)  {  if(B.income>E.income)  {  cout<<"\nHighest income:"<<B.name;  }  else if(B.income<E.income)  {  cout<<"\nHighest income:"<<E.name;  }  else  {  cout<<"\nBoth "<<B.name<<" and "<<E.name<<" have equal income:";  }  cout<<"\n------Checking who has highest employee------"<<endl;  if(B.people>10)  {  cout<<B.name<<" has more than 10 people involved in their business";  }  else  {  cout<<"\nNone";  }  }  void main()  {  Person P;  Businessman B;  Employee E;  clrscr();  cout<<"\nEntering values for Businessman..."<<endl;  B.get();  cout<<"\nEntering values for Employee..."<<endl;  E.get();  compare(B,E);  getch();  } **OUTPUT: C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (7).png** |
| **29.** | **Design a class media having data member’s title and publication. Derive a class book having a member to store number of pages in the book. Derive another class cd having a member to store the playing time. Define functions to read and display the data in the base class. Over-ride the same in both of the derived classes.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Media  {  public:  char title[50],pub[50];  virtual void read()=0;  virtual void display()=0;  };  class Book:public Media  {  public:  int pages;  void read()  {  cout<<"\nEnter Book title:";  cin>>title;  cout<<"\nEnter Publication:";  cin>>pub;  cout<<"\nEnter no of pages:";  cin>>pages;  }  void display()  {  cout<<"Book title:"<<title<<endl;  cout<<"Publication:"<<pub<<endl;  cout<<"No of pages:"<<pages<<endl;  }  };  class CD:public Media  {  public:  int time;  void read()  {  cout<<"\nEnter CD title:";  cin>>title;  cout<<"\nEnter Publication:";  cin>>pub;  cout<<"\nEnter playing time(in minutes):";  cin>>time;  }  void display()  {  cout<<"CD title:"<<title<<endl;  cout<<"Publication:"<<pub<<endl;  cout<<"Playing Time:"<<time<<endl;  }  };  void main()  {  Media \*ptr;  Book B;  CD C;  clrscr();  cout<<"------Enter values for Book------"<<endl;  ptr=&B;  ptr->read();  cout<<"------Enter values for CD------"<<endl;  ptr=&C;  ptr->read();  cout<<"------Book Details------"<<endl;  ptr=&B;  ptr->display();  cout<<"------CD Details------"<<endl;  ptr=&C;  ptr->display();  getch();  }  **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (8).png |
| 30. | **Create a class called 'TIME' that has three integer data members for hours, minutes and seconds constructor to initialize the object to zero constructor to initialize the object to some constant value member function to add two TIME objects member function to display time in HH:MM:SS format Write a main function to create two TIME objects, add them and display the result in HH:MM:SS format.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  class Time  {  int h,m,s;  public:  void get()  {  cout<<"Hours:";  cin>>h;  cout<<"Minutes:";  cin>>m;  cout<<"Seconds:";  cin>>s;  }  void sum(Time T1,Time T2)  {  s=T1.s + T2.s;  m=s/60;  s=s%60;  m=m + T1.m + T2.m;  h=m/60;  m=m%60;  h=h + T1.h + T2.h;  }  void display()  {  cout<<h<<":"<<m<<":"<<s;  }  };  void main()  {  clrscr();  Time T1,T2,T3;  cout<<"Enter 1st time T1:";  T1.get();  cout<<"\nEnter 2nd time T2:";  T2.get();  cout<<"\nTime T1:";  T1.display();  cout<<"\nTime T2:";  T2.display();  T3.sum(T1,T2);  cout<<"\nSum of T1 & T2:";  T3.display();  getch();  }  **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (9).png |
| **31.** | **a) Write a C++ program to implement basic stack operations using array.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  #include<stdlib.h>  #define stacksize 5  class stack  {  int top;  int s[stacksize];  public:  stack();  int isFull();  int isEmpty();  void push(int x);  void displaystack();  int pop();  int peep(int x);  void change(int x,int y);  };  stack::stack()  {  int i;  top=-1;  for(i=0;i<stacksize;i++)  s[i]=0;  }  int stack::isFull()  {  if(top>=stacksize-1)  return 1;  else  return 0;  }  int stack::isEmpty()  {  if(top<=-1)  return 1;  else  return 0;  }  void stack::push(int x)  {  if(isFull())  {  cout<<"\n Stack Overflow";  getch();  return;  }  top=top+1;  s[top]=x;  }  int stack::pop()  {  int x;  if(isEmpty())  {  cout<<"\nStack Underflow";  getch();  return 0;  }  x=s[top];  s[top]=0;  top--;  return x;  }  int stack::peep(int x)  {  if((top-x+1)<=-1)  cout<<"Stack is underflow";  else  return(s[top-x+1]);  }  void stack::change(int x,int y)  {  if((top-x+1)<=-1)  cout<<"Stack underflow";  else  s[top-x+1]=y;  }  void stack::displaystack()  {  int i;  for(i=0;i<stacksize;i++)  cout<<"\n"<<"s["<<i<<"]->"<<" "<<s[i];  }  void main()  {  int x,i,y;  int mchoice=1;  stack s;  while(mchoice!=0)  {  clrscr();  cout<<"\n\n";  cout<<" \*\*\*\*\*\* \*\*\*\* \*\*\*\*\* \*\*\*\*\*";  cout<<"\n 1->Push ";  cout<<"\n 2->Pop ";  cout<<"\n 3->Display ";  cout<<"\n 4->Peep ";  cout<<"\n 5->Change ";  cout<<"\n 0->Exit ";  cout<<" \*\*\*\*\* \*\*\*\* \*\*\*\*\* \*\*\*\*\*";  cout<<"\n\n ENter Your Choice:-";  cin>>mchoice;  switch(mchoice)  {  case 1:  cout<<"\n\n--------PUSH----------";  cout<<"\n Enter Element You want to Insert:-";  cin>>x;  s.push(x);  break;  case 2:  cout<<"\n\n---------POP-----------";  cout<<"\n The Deleted Element is:"<<s.pop();  break;  case 3:  cout<<"\n\n--------DISPLAY---------";  cout<<"\n =================";  s.displaystack();  cout<<"\n =================";  break;  case 4:  cout<<"\n\n--------PEEP-----";  cout<<"\n\n Enter Ith Element:-";  cin>>x;  cout<<"\n\nIth Element From Top of Stack is:-"<<s.peep(x);  break;  case 5:  cout<<"\n\n---------CHANGE-----";  cout<<"\n\n ENter Ith Element";  cin>>x;  cout<<"\n\nEnter New Element";  cin>>y;  s.change(x,y);  break;  case 0:  exit(1);  break;  }  cout<<"\nPress Any Key To Continue....";  getch();  }  getch();  }  **OUTPUT:-**  C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (13).pngC:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (12).pngC:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (11).pngC:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (10).png |
|  | **b) Write a C++ program to implement basic Queue operations using array.** |
|  | **SOLUTION:-** #include<iostream.h>  #include<conio.h>  #include<stdlib.h>  #define queuesize 5  class Queue  {  int q[queuesize],f,r,y;  public:  Queue();  void INSERT(int x);  int DELETE();  void DISPLAY();  };  Queue::Queue()  {  f=-1;  r=-1;  for(int i=0;i<queuesize;i++)  {  q[i]=0;  }  }  void Queue::INSERT(int x)  {  if(r>=queuesize-1)  {  cout<<"Queue Overflow"<<endl;  }  else  {  r=r+1;  q[r]=x;  if(f==-1)  { f=0;  }  }  }  int Queue::DELETE()  {  if(f==-1)  {  cout<<"Queue Underflow"<<endl;  }  else  {  y=q[f];  q[f]=0;  if(f==r)  {  f=-1;  r=-1;  }  else  {  f=f+1;  }  return y;  }  }  void Queue::DISPLAY()  {  for(int i=0;i<queuesize;i++)  {  cout<<"q["<<i<<"]";  cout<<" ";  }  cout<<endl;  cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;  for(i=0;i<queuesize;i++)  {  if(i>0)  {  cout<<" ";  }  cout<<" "<<q[i];  }  cout<<endl;  cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";  }  void main()  {  Queue Q;  int ch=1,x,y;  while(ch!=4)  {  clrscr();  cout<<"1. INSERT"<<endl;  cout<<”2. DELETE”<<endl;  cout<<"3. DISPLAY"<<endl;  cout<<"4. EXIT"<<endl;  cout<<"Enter your choice:"<<endl;  cin>>ch;  switch(ch)  {  case 1: cout<<"\n---INSERT---";  cout<<"\nEnter element you want to insert:";  cin>>x;  Q.INSERT(x);  break;  case 2: cout<<"\n---DELETE---";  y=Q.DELETE();  cout<<"\nDeleted Element:"<<y;  break;  case 3: cout<<"\n---DISPLAY---";  cout<<endl;  Q.DISPLAY();  break;  case 4: exit(1);  getch();  }  cout<<"\nPress any key to continue";  getch();  }  getch();  }  **OUTPUT:** C:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (16).pngC:\Users\IT STUDENT\Pictures\Screenshots\Screenshot (15).png |

**SUBMISSION DATE:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 FACULTY SIGNATURE  
(Dr. Niky K. Jain)**