# COMPUTER SCIENCE

## LAB PRACTICALS

2019-20

LANGUAGE C++

NAME

HARSH MISHRA

CLASS

XII – D

# LAB PRACTICAL 1

**Q-1**
**Write a function to check whether a number is prime or not.**
**Now use this function to display prime numbers upto a number.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to check whether a number is prime or not or tell about number of prime
numbers upto a number.
#include<iostream>
using namespace std;
void opt1();                        void opt2();
char isprime(int num);              void n_prime(int);
int main()
	{
		int n;char c;
		do
			{
				cout<<"\nWhat you want to do?\n"
					<<"1.Check whether is prime?\n"
					<<"2.Get prime numbers upto a number?\n";
				cout<<"\nEnter your choice : ";
				cin>>n;
				if(n==1)
					opt1();
				else if(n==2)
					opt2();
				else
					cout<<"\n|Wrong Input|\n";
				cout<<"\nDo you want to continue(Y/N)? : ";
				cin>>c;
			}while(c=='Y'||c=='y');
	}
void opt1()
	{
		int num;char ans;
		cout<<"\nEnter the number to be checked : ";
		cin>>num;
		if(num==1)
		{
			cout<<"Number you have entered is neither prime nor composite.";
			exit(0);
		}
		ans=isprime(num);
		if(ans=='Y')
			cout<<"The number you have entered is prime.\n";
		else
			cout<<"The number you have entered is composite.\n";
	}

void opt2()
```

```
	{
		int n;
		cout<<"\nEnter the ending point : ";
		cin>>n;
		n_prime(n);
	}
char isprime(int num)
	{
		int i,nf=0;
		for(i=1;i<=num;++i)
			if(num%i==0)
				++nf;
		if(nf==2)
			return 'Y';
		else
			return 'N';
	}
void n_prime(int n)
	{
		int i,i2,nf=0;
		cout<<"|Below are the prime numbers b/w these numbers|\n";
		for(i=0;i<n;++i)
			{
				for(i2=i;i2;--i2)
					if(i%i2==0)
						++nf;
				if(nf==2)
					cout<<i<<"_";
				nf=0;
			}
		cout<<endl;
	}
```

**|OUTPUT IS ON THE NEXT PAGE|**

```
What you want to do?
1.Check whether is prime?
2.Get prime numbers upto a number?

Enter your choice : 1

Enter the number to be checked : 1
Number you have entered is neither prime nor composite.

Do you want to continue(Y/N)? : y

What you want to do?
1.Check whether is prime?
2.Get prime numbers upto a number?

Enter your choice : 1

Enter the number to be checked : 43
The number you have entered is prime.

Do you want to continue(Y/N)? : y

What you want to do?
1.Check whether is prime?
2.Get prime numbers upto a number?

Enter your choice : 1

Enter the number to be checked : 78
The number you have entered is composite.

Do you want to continue(Y/N)? : y

What you want to do?
1.Check whether is prime?
2.Get prime numbers upto a number?

Enter your choice : 2

Enter the ending point : 23
|Below are the prime numbers b/w these numbers|
2_3_5_7_11_13_17_19_

Do you want to continue(Y/N)? : n

--------------------------------
Process exited after 41.11 seconds with return value 0
Press any key to continue . . .
```

**Q-2**
**Write a function to check whether a string is palindrome or not. The function will return 'y' if the passed string is a palindrome otherwise 'n' will be returned.**
**Now write main() function to declare and obtain a string from the user. Call the above function ( pass the string as an argument) and display the result.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to check whether a function is pallindrome or not(XII revised).
#include<iostream>
#include<string.h>
#include<stdio.h>
using namespace std;
char ispallin(char[]);
int main()
	{
		int res;	char s[10];
		cout<<"Enter the string to be checked : ";	gets(s);
		res=ispallin(s);
		if(res=='Y')
			cout<<"Yes, the string is pallindrome.";
		else
			cout<<"No, the string is not pallindrome.";
	}
char ispallin(char s[10])
	{
		int o;		char r_s[10];
		strcpy(r_s,s);	strrev(r_s);
		o=strcmp(r_s,s);
		if(o==0)
			return 'Y';
		else
			return 'N';
	}
```

```
Enter the string to be checked(case sensitive): malayalam
Yes, the string is pallindrome.
--------------------------------
Process exited after 6.811 seconds with return value 0
Press any key to continue . . .
```

```
Enter the string to be checked(case sensitive): curtain
No, the string is not pallindrome.
--------------------------------
Process exited after 29.87 seconds with return value 0
Press any key to continue . . .
```

**Q-3**
**Write a function to add elements of diagonals of square array entered by user.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to add values at diagonals of square array.
#include<iostream>
void add(int[][5],int);
int main()
	{
		int a[5][5],i,j,sz,sum1=0,sum2=0;
		cout<<"Enter the size of side of square matrix : ";	cin>>sz;
		cout<<"Enter the array elements : \n";
		for(i=0;i<sz;++i)	for(j=0;j<sz;++j)		cin>>a[i][j];
		cout<<"Elements entered by you : \n";
		for(i=0;i<sz;++i)
			{	for(j=0;j<sz;++j)
					cout<<a[i][j]<<"\t";
				cout<<endl;					}
		add(a,sz);
	}
void add(int a[][5],int sz)
	{
		int i,j,sum1=0,sum2=0;
		for(i=0;i<sz;++i)
			for(j=0;j<sz;++j)
				{	if(i==j)
						sum1+=a[i][j];
					if(i+j==sz-1)
						sum2+=a[i][j];		}
		cout<<"\nSum of the elements at primary diagonal is : "<<sum1;
		cout<<"\nSum of the elements at secondary diagonal is : "<<sum2;
	}
```

```
Enter the size of side of square matrix : 3
Enter the array elements :
 •
 •
 •
Elements entered by you :
12      34      51
22      19      33
41      39      47

Sum of the elements at primary diagonal is : 78
Sum of the elements at secondary diagonal is : 111
--------------------------------
Process exited after 35.21 seconds with return value 0
Press any key to continue . . .
```

**Q-4**
**Write a function to check which time is greater.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to check which time is greater.
#include<iostream>
struct time
	{	int hr,mn,ss;	};
time i_t;
time gtr(time,time);	int chk(time,time);
void enter(time&);	void display(time);
int main()
	{
		time t1,t2,t;
		cout<<"Enter value of time 1 : \n";	enter(t1);
		cout<<"Enter value of time 2 : \n";	enter(t2);
		t=gtr(t1,t2);
		if(chk(t,i_t))
			cout<<"Times are equal.";
		else
			{	cout<<"Greater time is : \n";
				display(t);			}
	}
int chk(time t,time i_t)
	{
		if(t.hr==i_t.hr)if(t.mn==i_t.mn)if(t.ss==i_t.ss)
								return 1;
		else
			return 0;
	}
void enter(time &t)
	{
		cout<<"Enter it's hour : ";		cin>>t.hr;
		cout<<"Enter it's minutes : ";	cin>>t.mn;
		cout<<"Enter it's seconds : ";	cin>>t.ss;
	}
void display(time t)
	{
		cout<<"Hour : "<<t.hr;
		cout<<"\nMinutes : "<<t.mn;
		cout<<"\nSeconds : "<<t.ss;
	}


time gtr(time t1,time t2)
	{
		if(t1.hr>t2.hr||t1.hr<t2.hr)
			if(t1.hr>t2.hr)	return t1;
			else		return t2;
		else if(t1.mn>t2.mn||t1.mn<t2.mn)
			if(t1.mn>t2.mn)		return t1;
```

```
            else                    return t2;
        else if(t1.ss>t2.ss||t1.ss<t2.ss)
            if(t1.ss>t2.ss)         return t1;
            else                    return t2;
        else                        return i_t;
    }
```

```
Enter value of time 1 :
Enter it's hour : 4
Enter it's minutes : 55
Enter it's seconds : 32
Enter value of time 2 :
Enter it's hour : 1
Enter it's minutes : 12
Enter it's seconds : 56
Greater time is :
Hour : 4
Minutes : 55
Seconds : 32
--------------------------------
Process exited after 27.17 seconds with return value 0
Press any key to continue . . .
```

```
Enter value of time 1 :
Enter it's hour : 2
Enter it's minutes : 44
Enter it's seconds : 22
Enter value of time 2 :
Enter it's hour : 2
Enter it's minutes : 44
Enter it's seconds : 22
Times are equal.
--------------------------------
Process exited after 6.817 seconds with return value 0
Press any key to continue . . .
```

# LAB PRACTICAL 2

**Q-1**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Classes and Objects #04
#include<iostream>
#include<stdio.h>
using namespace std;
class flight
	{
		int fno;
		char dest[30];
		float dist;
		float fuel;
		void calfuel();
	public:
		void feedinfo();
		void showinfo();
	};
void flight::feedinfo()
	{
		cout<<"Enter the flight number : ";	cin>>fno;
		cout<<"Enter the destination : ";	cin>>dest;
		cout<<"Enter the distance : ";		cin>>dist;
		calfuel();
	}
void flight::calfuel()
	{
		if(dist<=1000)
			fuel=500;
		else if(dist>1000&&dist<=2000)
			fuel=1100;
		else if(dist>2000)
			fuel=2200;
	}
void flight::showinfo()
	{
		cout<<"\n|Following are the details of the flight|"
			<<"\nFlight number : "<<fno
			<<"\nDestination : "<<dest
			<<"\nDistance travelled : "<<dist
			<<"\nFuel used : "<<fuel;
	}
int main()
	{
		flight f;
		f.feedinfo();
		f.showinfo();
	}
```

```
Enter the flight number : 32166
Enter the destination : Mumbai
Enter the distance : 675

|Following are the details of the flight|
Flight number : 32166
Destination : Mumbai
Distance travelled : 675
Fuel used : 500
--------------------------------
Process exited after 15.15 seconds with return value 0
Press any key to continue . . .
```

**Q-2**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Classes and objects #07
#include<iostream>
#include<string.h>
using namespace std;
class garments
	{
		char gcode[20],gtype[20];
		int gsize;
		char gfabric[20];
		float gprice;
		void assign();
	public:
		garments();
		garments(char[],int);
		void input();
		void display();
	};
garments::garments()
	{
		strcpy(gcode,"NOT ALLOTED");
		strcpy(gtype,gcode);
		strcpy(gfabric,gtype);
		gprice=gsize=0;
	}
garments::garments(char x[],int y)
	{
		strcpy(gcode,x);
		strcpy(gtype,gcode);
		strcpy(gfabric,gtype);
```

```
            gprice=gsize=y;
      }
void garments::input()
      {
            cout<<"Enter the garment code : ";
            cin>>gcode;
            cout<<"Enter the garment type(Trouser or Shirt) : ";
            cin>>gtype;
            cout<<"Enter the garment size : ";
            cin>>gsize;
            cout<<"Enter the garment fabric(Cotton or Other) : ";
            cin>>gfabric;
            assign();
      }
void garments::assign()
      {
            if(strcmpi(gfabric,"Cotton")==0)
                  if(strcmpi(gtype,"Trouser")==0)
                        gprice=1300;
                  else if(strcmpi(gtype,"Shirt")==0)
                        gprice=1100;
                  else
                        strcpy(gtype,"Wrong Input");

            else
                  if(strcmpi(gtype,"Trouser")==0)
                        gprice=1300-130;
                  else if(strcmpi(gtype,"Shirt")==0)
                        gprice=1100-110;
                  else
                        strcpy(gtype,"Wrong Input");
      }
void garments::display()
      {
            cout<<"\nGarment code : "<<gcode
                  <<"\nGarment type : "<<gtype
                  <<"\nGarment size : "<<gsize
                  <<"\nGarment fabric : "<<gfabric
                  <<"\nGarment price : "<<gprice;
      }
int main()
      {
            garments g1,g2("INITIALIZED",500),g3;
            cout<<"|Details of garment not yet initialized|";
            g1.display();
            cout<<"\n|Details of garment initialized by me|";
            g2.display();
            cout<<"\n|Enter the details of your garment|\n";
            g3.input();
            cout<<"|Details of garment entered by me|";
            g3.display();                                          }
```

```
|Details of garment not yet initialized|
Garment code : NOT ALLOTED
Garment type : NOT ALLOTED
Garment size : 0
Garment fabric : NOT ALLOTED
Garment price : 0
|Details of garment initialized by me|
Garment code : INITIALIZED
Garment type : INITIALIZED
Garment size : 500
Garment fabric : INITIALIZED
Garment price : 500
|Enter the details of your garment|
Enter the garment code : 54388
Enter the garment type(Trouser or Shirt) : Shirt
Enter the garment size : 42
Enter the garment fabric(Cotton or Other) : other
|Details of garment entered by me|
Garment code : 54388
Garment type : Shirt
Garment size : 42
Garment fabric : other
Garment price : 990
--------------------------------
Process exited after 57.42 seconds with return value 0
Press any key to continue . . .
```

# LAB PRACTICAL 3

**Q-1**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to enter details of the books using inheritance.
#include<iostream>
using namespace std;
class publication
	{
		char title[10];
		float price;
			public:
		void getdata();
		void putdata();
	};
class book:publication
	{
		int pg_cnt;
			public:
		void getdata();
		void putdata();
	};
class tape:publication
	{
		float plt;
			public:
		void getdata();
		void putdata();
	};
void publication::getdata()
	{
		cout<<"Enter the title : ";		cin>>title;
		cout<<"Enter the price : ";		cin>>price;
	}
void publication::putdata()
	{
		cout<<"Title : "<<title;
		cout<<"\nPrice : "<<price;
	}
void book::getdata()
	{
		publication::getdata();
		cout<<"Enter the no. of pages : ";	cin>>pg_cnt;
	}
void book::putdata()
	{
		publication::putdata();
		cout<<"\nNo. of pages : "<<pg_cnt;
	}
void tape::getdata()
```

```
	{
		publication::getdata();
		cout<<"Enter playing time : ";		cin>>plt;
	}
void tape::putdata()
	{
		publication::putdata();
		cout<<"\nPlaying time : "<<plt;
	}
int main()
	{
		book b;tape t;
		cout<<"|Enter the details of the book|\n";
		b.getdata();
		cout<<"|Enter the details of the tape|\n";
		t.getdata();
		cout<<"\nDetails of the book|\n";
		b.putdata();
		cout<<"\nDetails of the tape|\n";
		t.putdata();
	}
```

```
|Enter the details of the book|
Enter the title : Rookie
Enter the price : 320
Enter the no. of pages : 566
|Enter the details of the tape|
Enter the title : Cello
Enter the price : 35
Enter playing time : 345

Details of the book|
Title : Rookie
Price : 320
No. of pages : 566
Details of the tape|
Title : Cello
Price : 35
Playing time : 345
--------------------------------
Process exited after 54.35 seconds with return value 0
Press any key to continue . . .
```

**Q-2**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program for entering details of the student using inhertance.
#include<iostream>
#include<string.h>
using namespace std;
class student
	{
		char name[15];
		int rno;
			protected:
		int cls;
			public:
		student()
			{
				strcpy(name,"NOT ASSIGNED");
				rno=0;cls=0;
			}
		void input()
			{
				cout<<"Enter the name of the student : ";
				cin>>name;
				cout<<"Enter the roll no. of the student : ";
				cin>>rno;
				cout<<"Enter the class of the student : ";
				cin>>cls;
			}
		void display()
			{
				cout<<"Name of the student : "<<name;
				cout<<"\nRoll no. of the student : "<<rno;
				cout<<"\nClass of the student : "<<cls;
			}
	};
class test:public student
	{
		float eng,phy,chem,math,cs;
		void caltotal()
			{
				total=eng+phy+chem+math+cs;
			}
			protected:
		float total;
			public:
		test()
			{
				eng=phy=chem=math=cs=0;
			}
		void getdata()
			{
```

```
                student::input();
                cout<<"|Enter marks of the student|\n";
                cout<<"English:";cin>>eng;
                cout<<"Physics:";cin>>phy;
                cout<<"Chemistry:";cin>>chem;
                cout<<"Maths:";cin>>math;
                cout<<"Computer Science:";cin>>cs;
                caltotal();
        }
    void putdata()
        {
                student::display();
                cout<<"\n|Marks of the student|\n";
                cout<<"English:"<<eng;
                cout<<"\nPhysics:"<<phy;
                cout<<"\nChemistry:"<<chem;
                cout<<"\nMaths:"<<math;
                cout<<"\nComputer Science:"<<cs;
                cout<<"\n\nTotal Marks:"<<total;
        }
};
class result:public test
{
    float per;
    char grd;
        public:
    result()
        {
                per=0;grd='E';
        }
    void calper()
        {
                per=total/5;
        }
    void calgrd()
        {
                if(per>=90)
                        grd='A';
                else if(per>=75&&per<90)
                        grd='B';
                else if(per>=60&&per<75)
                        grd='C';
                else if(per>=40&&per<60)
                        grd='D';
                else
                        grd='E';
        }
    void get_result()
        {
                test::getdata();
                calper();calgrd();
        }
```

```
        void dis_result()
            {
                test::putdata();
                cout<<"\n\n|Result of the student is as follows|";
                cout<<"\nPercentage:"<<per;
                cout<<"\nGrade:"<<grd;
            }
    };
int main()
    {
        result r;
        cout<<"|Enter the details of the student|\n";          r.get_result();
        cout<<"\n|Details of the student are as follows|\n";   r.dis_result();
    }
```

```
|Enter the details of the student|
Enter the name of the student : Dhruv
Enter the roll no. of the student : 27
Enter the class of the student : 11
|Enter marks of the student|
English:89
Physics:94
Chemistry:93
Maths:96
Computer Science:95

|Details of the student are as follows|
Name of the student : Dhruv
Roll no. of the student : 27
Class of the student : 11
|Marks of the student|
English:89
Physics:94
Chemistry:93
Maths:96
Computer Science:95

Total Marks:467

|Result of the student is as follows|
Percentage:93.4
Grade:A
--------------------------------
Process exited after 17.52 seconds with return value 0
Press any key to continue . . .
```

# LAB PRACTICAL 4

**Program to perform many task on a binary file.**

**[Before the main program]**

**<adcls.h>**

```
class student
{
        int rollno;
        char name[20];
        float per;
    public:
        void getdata();
        void showdata();
        void newnameper();
        int retroll()
            { return rollno; }
        float retper()
            { return per; }
        char * retname()
            { return name; }
};
void student::getdata()
    {
        cout<<"\nEnter Roll no. : ";
        cin>>rollno;
        cout<<"Enter the name : ";
        cin.ignore();
        gets(name);
        cout<<"Enter percentage : ";
        cin>>per;
    }
void student::showdata()
    {
        cout<<"\nRoll no. : "<<rollno;
        cout<<"\nName : "<<name;
        cout<<"\nPercentage : "<<per;
    }
void student::newnameper()
    {
        cout<<"\nEnter new name : ";
        cin.ignore();
        gets(name);
        cout<<"Enter new per : ";
        cin>>per;
    }
```

**<adfnt.h>**

```
void ent_data()
	{
		ofstream f;
		f.open("record.txt",ios::out|ios::binary);
		student s;char c;
		cout<<"Enter the details of the students : \n";
		do
			{
				s.getdata();
				f.write((char*)&s,sizeof(s));
				cout<<"\nDo you want to continue : ";
				cin>>c;
			}while(c=='y'||c=='Y');
		f.close();
	}
void shw_data()
	{
		ifstream f;
		student s;
		f.open("record.txt",ios::in|ios::binary);
		cout<<"\nDetails of the students are as follows : \n";
		f.read((char*)&s,sizeof(s));
		while(!f.eof())
			{
				s.showdata();
				f.read((char*)&s,sizeof(s));
			}
		f.close();
	}
```

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to perform many task in binary file.
#include<fstream>
#include<iostream>
#include<stdio.h>
using namespace std;
#include<adcls.h>
#include<adfnt.h>
void no_rec()
	{
		ifstream f;
		f.open("record.txt",ios::in|ios::binary);
		student s;int cnt=0;
		f.read((char*)&s,sizeof(s));
		while(!f.eof())
			{
				++cnt;
				f.read((char*)&s,sizeof(s));
```

```
                }
        cout<<"\nNo. of records in this file : "<<cnt<<endl;
        f.close();
    }
void del_rec()
    {
        ifstream f1;
        f1.open("record.txt",ios::in|ios::binary);
        ofstream f2;
        f2.open("temp.txt",ios::out|ios::binary);
        student s;int rno;int flag=0;
        cout<<"\nEnter the roll no. of the student : ";
        cin>>rno;
        f1.read((char*)&s,sizeof(s));
        while(!f1.eof())
            {
                if(s.retroll()==rno)
                    flag=1;
                if(s.retroll()!=rno)
                    f2.write((char*)&s,sizeof(s));
                f1.read((char*)&s,sizeof(s));
            }
        if(flag==1)
            cout<<"\nRecord successfully deleted.\n";
        else
            cout<<"\nNo such record found\n";
        f1.close();
        f2.close();
        remove("record.txt");
        rename("temp.txt","record.txt");
    }
void modify()
    {
        fstream f;
        f.open("record.txt",ios::in|ios::out|ios::ate|ios::binary);
        f.seekg(0);
        student s;int rno;int flag=0;
        cout<<"\nEnter the roll no. of the student : ";
        cin>>rno;
        f.read((char*)&s,sizeof(s));
        while(!f.eof())
            {
                if(s.retroll()==rno)
                    {
                        s.newnameper();
                        f.seekp((-1)*sizeof(s),ios::cur);
                        f.write((char*)&s,sizeof(s));
                        flag=1;
                    }
                f.read((char*)&s,sizeof(s));
            }
        if(flag==1)
```

```
                cout<<"\nRecord successfully modified\n";
        else
                cout<<"\nNo such record found\n";
        f.close();
    }
void ad_data()
    {
        ofstream f;
        f.open("record.txt",ios::out|ios::app|ios::binary);
        student s;char c;
        cout<<"\nEnter the details of new students : \n";
        do
            {
                s.getdata();
                f.write((char*)&s,sizeof(s));
                cout<<"\nDo you want to continue : ";
                cin>>c;
            }while(c=='y'||c=='Y');
        f.close();
    }
void disp_rec()
    {
        ifstream f;
        f.open("record.txt",ios::in|ios::binary);
        student s;float n;
        cout<<"\nEnter the percentage criteria : ";
        cin>>n;
        cout<<"\nDetails of the students is as follows : \n";
        f.read((char*)&s,sizeof(s));
        while(!f.eof())
            {
                if(s.retper()>n)
                    s.showdata();
                f.read((char*)&s,sizeof(s));
            }
        f.close();
    }
int main()
    {
        cout<<"---->ENTER MAIN DATA\n\n";
        ent_data();
        cout<<"\n---->NO. OF RECORD\n";
        no_rec();
        cout<<"\n---->DELETION OF A RECORD\n";
        del_rec();
        cout<<"\n---->MODIFICATION OF A RECORD\n";
        modify();
        cout<<"\n---->ADD A RECORD\n";
        ad_data();
        cout<<"\n---->FINAL RECORD\n";
        disp_rec();                                    }
```

```
---->ENTER MAIN DATA

Enter the details of the students :

Enter Roll no. : 1
Enter the name : Suresh
Enter percentage : 93

Do you want to continue : y

Enter Roll no. : 2
Enter the name : Ganesh
Enter percentage : 94

Do you want to continue : y

Enter Roll no. : 3
Enter the name : Yogesh
Enter percentage : 96

Do you want to continue : n
```

```
---->NO. OF RECORD

No. of records in this file : 3
```

```
---->DELETION OF A RECORD

Enter the roll no. of the student : 2

Record successfully deleted.
```

```
---->MODIFICATION OF A RECORD

Enter the roll no. of the student : 1

Enter new name : Naagesh
Enter new per : 93.5

Record successfully modified
```

```
---->ADD A RECORD

Enter the details of new students :

Enter Roll no. : 4
Enter the name : Ramesh
Enter percentage : 89.8

Do you want to continue : y

Enter Roll no. : 5
Enter the name : Rakesh
Enter percentage : 92.3

Do you want to continue : n
```

```
---->FINAL RECORD

Enter the percentage criteria : 0

Details of the students is as follows :

Roll no. : 1
Name : Naagesh
Percentage : 93.5
Roll no. : 3
Name : Yogesh
Percentage : 96
Roll no. : 4
Name : Ramesh
Percentage : 89.8
Roll no. : 5
Name : Rakesh
Percentage : 92.3
```

```
--------------------------------
Process exited after 137.6 seconds with return value 0
Press any key to continue . . .
```

# LAB PRACTICAL 5

**Q-1**

**Program using pointers to find the square of even numbers and cube of odd numbers present in an array.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to find square and cube of even and odd no. respectively in array by pointers.
#include<iostream>
#include<math.h>
using namespace std;
int main()
	{
		int *p;
		int n,i;
		cout<<"Enter the size of the array : ";
		cin>>n;
		p=new int[n];
		cout<<"Enter the pointer array elements : \n";
		for(i=0;i<n;++i)
			cin>>*(p+i);
		cout<<"Pointer array elements entered by you are : \n";
		for(i=0;i<n;++i)
			cout<<*(p+i)<<"\t";
		cout<<"\nOutput in the desired way : \n";
		for(i=0;i<n;++i)
			{
				if(*(p+i)%2==0)
					cout<<pow(*(p+i),2)<<"\t";
				else
					cout<<pow(*(p+i),3)<<"\t";
			}
		delete[] p;
	}
```

```
Enter the size of the array : 4
Enter the pointer array elements :
9
6
2
8
Pointer array elements entered by you are :
9       6       2       8
Output in the desired way :
729     36      4       64
--------------------------------
Process exited after 10.43 seconds with return value 0
Press any key to continue . . .
```

**Q-2**
**Write a program to find whether a string is palindrome or not using pointers.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to check whether a string is pallindrome or not using pointer.
#include<iostream>
#include<stdio.h>
#include<string.h>
using namespace std;
int main()
	{
		char str[10];
		cout<<"Enter the string : ";
		gets(str);
		int len;
		len=strlen(str);
		char *p;
		p=str;
		cout<<"String entered by you : ";
		cout<<p<<endl;
		int i,k,flag=0;
		for(i=0,k=len-1;i<len/2;++i,--k)
			if(*(p+i)!=*(p+k))
				flag=1;
		if(flag==1)
			cout<<"String is not pallindrome.";
		else
			cout<<"String is pallindrome.";
	}
```

```
Enter the string : malayalam
String entered by you : malayalam
String is pallindrome.
--------------------------------
Process exited after 10.7 seconds with return value 0
Press any key to continue . . .
```

```
Enter the string : rookie
String entered by you : rookie
String is not pallindrome.
--------------------------------
Process exited after 3.659 seconds with return value 0
Press any key to continue . . .
```

**Q-3**

**Write a program to create an integer array of n elements dynamically and display the highest & lowest values with their positions. Also delete the entire array.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to arrange highest and lowest value of integer array created dynamically using pointer.
#include<iostream>
using namespace std;
int main()
	{
		int n,i,j,temp;
		cout<<"Enter the size of the array : ";
		cin>>n;
		int *p=new int[n];
		cout<<"Enter the pointer array elements : \n";
		for(i=0;i<n;++i)
			cin>>*(p+i);
		cout<<"Pointer array elements entered by you : \n";
		for(i=0;i<n;++i)
			cout<<*(p+i)<<"\t";
		int h,l;
		h=l=*p;
		for(i=0;i<n;++i)
			{
				if(*(p+i)>h)
					h=*(p+i);
				if(*(p+i)<l)
					l=*(p+i);
			}
		cout<<endl<<"Highest value : "<<h;
		cout<<endl<<"Lowest value : "<<l;
		for(i=0;i<n-1;++i)
			for(j=0;j<n-1-i;++j)
				{
					if(*(p+j)>*(p+j+1))
						{
							temp=*(p+j);
							*(p+j)=*(p+j+1);
							*(p+j+1)=temp;
						}
				}
		cout<<"\nOutput in the desired way : \n";
		for(i=0;i<n;++i)
			cout<<*(p+i)<<"\t";
		delete[] p;
	}
```

```
Enter the size of the array : 6
Enter the pointer array elements :
 •
 •
 •
Pointer array elements entered by you :
12      32      22      41      29      26
Highest value : 41
Lowest value : 12
Output in the desired way :
12      22      26      29      32      41
--------------------------------
Process exited after 13.9 seconds with return value 0
Press any key to continue . . .
```

**Q-4**

**Create dynamically an array of n elements of structure GAME. Obtain data of the array elements from the user and display all the data. Also display the details of the Player who scored highest points.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to create a structure object dynamically and then fill and display its values
using pointer.
#include<iostream>
using namespace std;
struct game
      {
            int pno;
            char pname[10];
            float points;
      }*p;
int main()
      {
            int n,i;
            cout<<"Enter number of player : ";
            cin>>n;
            p=new game[n];
            for(i=0;i<n;++i)
                  {
                        cout<<"Enter the details of the player "<<i+1<<": \n";
                        cout<<"Enter player number : ";cin>>(p+i)-
>pno;cin.ignore();
                        cout<<"Enter player name : ";gets((p+i)->pname);
                        cout<<"Enter player points : ";cin>>(p+i)->points;
                  }
            for(i=0;i<n;++i)
                  {
                        cout<<"\nDetails of the player "<<i+1<<": ";
                        cout<<"\nPlayer number : "<<(p+i)->pno;
```

```
                        cout<<"\nPlayer name : "<<(p+i)->pname;
                        cout<<"\nPlayer points : "<<(p+i)->points;
                }
        float h1=p->points;int h2;
        for(i=0;i<n;++i)
                if((p+i)->points>h1)
                        {
                                h1=(p+i)->points;
                                h2=i;
                        }
        cout<<"\n\nAnd player "<<h2+1<<" has the highest points.";
}
```

```
Enter number of player : 4
Enter the details of the player 1:
Enter player number : 32155
Enter player name : Raman
Enter player points : 665
Enter the details of the player 2:
Enter player number : 44522
Enter player name : Tarun
Enter player points : 541
Enter the details of the player 3:
Enter player number : 35590
Enter player name : Akshay
Enter player points : 437
Enter the details of the player 4:
Enter player number : 54431
Enter player name : Rajiv
Enter player points : 678

Details of the player 1:
Player number : 32155
Player name : Raman
Player points : 665
Details of the player 2:
Player number : 44522
Player name : Tarun
Player points : 541
Details of the player 3:
Player number : 35590
Player name : Akshay
Player points : 437
Details of the player 4:
Player number : 54431
Player name : Rajiv
Player points : 678

And player 4 has the highest points.
--------------------------------
Process exited after 80.72 seconds with return value 0
Press any key to continue . . .
```

# LAB PRACTICAL 6

**Q-1**
**Program to perform multiple task on 2D array.**

**|PROGRAMMING and OUTPUT|**

```
 //DEV C++
//Program to perform multiple task with 2D array.
#include<iostream>
using namespace std;
int sum_alt(int[][10],int,int);
void swapclm(int[][10],int,int);
void sum_row(int[][10],int,int);
int main()
	{
		int a[10][10],row,clm,i,j,sum;
		cout<<"Enter the size of the row : ";
		cin>>row;
		cout<<"Enter the size of the column : ";
		cin>>clm;
		cout<<"Enter the array elements : \n";
		for(i=0;i<row;++i)
			for(j=0;j<clm;++j)
				cin>>a[i][j];
		cout<<"\nArray entered by you : \n";
		for(i=0;i<row;++i)
			{
				for(j=0;j<clm;++j)
					cout<<a[i][j]<<"\t";
				cout<<endl;
			}
		sum=sum_alt(a,row,clm);
		cout<<"\nSum of alternate elements is : "<<sum;
		swapclm(a,row,clm);
		cout<<"\n\nDesired array after swapping : \n";
		for(i=0;i<row;++i)
			{
				for(j=0;j<clm;++j)
					cout<<a[i][j]<<"\t";
				cout<<endl;
			}
		cout<<"\nSum of each row are as follows : \n";
		sum_row(a,row,clm);
	}
int sum_alt(int a[][10],int row,int clm)
	{
		int i,j,k=0,b[35],sz,sum=0;
		sz=row*clm;
		for(i=0;i<row;++i)
			for(j=0;j<clm;++j)
				{
					b[k]=a[i][j];
```

```
                        ++k;
                    }
            for(i=0;i<sz;i+=2)
                sum+=b[i];
            return sum;
        }
void swapclm(int a[][10],int row,int clm)
        {
            int i,temp;
            for(i=0;i<row;++i)
                {
                    temp=a[i][0];
                    a[i][0]=a[i][clm-1];
                    a[i][clm-1]=temp;
                }
        }
void sum_row(int a[][10],int row,int clm)
        {   int i,j,sum_r;
            for(i=0;i<row;++i)
                {   sum_r=0;
                    for(j=0;j<clm;++j)
                        sum_r+=a[i][j];
                    cout<<"Sum of row "<<i+1<<" is : "<<sum_r<<endl;    }}
```

```
Enter the size of the row : 2
Enter the size of the column : 3
Enter the array elements :
•
•
•
Array entered by you :
12      33      22
11      10      9

Sum of alternate elements is : 44

Desired array after swapping :
22      33      12
9       10      11

Sum of each row are as follows :
Sum of row 1 is : 67
Sum of row 2 is : 30

--------------------------------
Process exited after 24.23 seconds with return value 0
Press any key to continue . . .
```

**Q-2**

**Program to merge two array, one ascending and one descending.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to merge two array, one ascending and one descending.
#include<iostream>
using namespace std;
void insertion_sort(int x[],int sz);                void selection_sort(int x[],int sz);
void merge(int a[],int m,int b[],int n,int c[],int k);
int main()
	{
		int a[10],m,b[10],n,c[20],k,i;
		cout<<"Enter the size of array 1 : ";		cin>>m;
		cout<<"Enter the array 1 elements : \n";
		for(i=0;i<m;++i)				cin>>a[i];
		cout<"|Arranging in ascending order|\n";
		insertion_sort(a,m);
		cout<<"Enter the size of array 2 : ";		cin>>n;
		cout<<"Enter the array 2 elements : \n";
		for(i=0;i<n;++i)				cin>>b[i];
		cout<"|Arranging in descending order|\n";
		selection_sort(b,n);
		cout<<"Desired array 1 : \n";
		for(i=0;i<m;++i)				cout<<a[i]<<"\t";
		cout<<"\nDesired array 2 : \n";
		for(i=0;i<n;++i)				cout<<b[i]<<"\t";
		k=m+n;
		merge(a,m,b,n,c,k);
		cout<<"\nElements of merged array are : \n";
		for(i=0;i<k;++i)				cout<<c[i]<<"\t";
	}
void insertion_sort(int x[],int sz)
	{
		int i,j,temp;
		for(i=1;i<sz;++i)
			{
				temp=x[i];
				j=i-1;
				while((temp<x[j])&&(j>=0))
					{
						x[j+1]=x[j];
						--j;
					}
				x[j+1]=temp;
			}
	}
void selection_sort(int x[],int sz)
	{
		int i,j,large,pos,temp;
		for(i=0;i<sz;++i)
			{
```

```
                        large=x[i];
                        pos=i;
                        for(j=i+1;j<sz;++j)
                              {
                                    if(x[j]>large)
                                          {
                                                large=x[j];
                                                pos=j;
                                          }
                              }
                        temp=x[i];
                        x[i]=x[pos];
                        x[pos]=temp;
                  }
      }
void merge(int a[],int m,int b[],int n,int c[],int k)
      {
            int i,j,o;
            i=o=0;j=n-1;
            while(i<m&&j>=0)
                  if(a[i]<b[j])       c[o++]=a[i++];
                  else                c[o++]=b[j--];
            while(i<m)        c[o++]=a[i++];
            while(j>=0)       c[o++]=b[j--];
      }
```

```
Enter the size of array 1 : 4
Enter the array 1 elements :
 •
 •
 •
Enter the size of array 2 : 3
Enter the array 2 elements :
 •
 •
 •
Desired array 1 :
5       12      15      19
Desired array 2 :
17      11      4
Elements of merged array are :
4       5       11      12      15      17      19
--------------------------------
Process exited after 37.98 seconds with return value 0
Press any key to continue . . .
```

# LAB PRACTICAL 7

**Q-1**
**Write a menu driven program from static implementation of STACK.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to push,pop,display in stack.
#include<iostream>
#include<process.h>
#define size 5
using namespace std;
class stack
	{		int a[size];		int top;
		public:
			stack()
				{	top=-1;	}
			void push(int);		int pop();		void display();		};
void stack::push(int item)
	{	if(top==size-1)		cout<<"|Overflow|\n";
		else
			{	++top;
				a[top]=item;	}		}
int stack::pop()
	{	if(top==-1)
			{	cout<<"|Underflow|\n";
				return -1;		}
		else
			{	int item;
				item=a[top];
				top--;
				return item;	}		}
void stack::display()
	{	int i;
		if(top==-1)
			{	cout<<"|Empty stack|\n";
				return;		}
		cout<<"|Stack is as follows|\n";
		for(i=top;i>=0;--i)		cout<<a[i]<<"\t";
		cout<<endl;		}
int main()
	{	stack s;	int c,v;
		do
		{	cout<<"\n_MENU_\n";
			cout<<"1.Push"
				<<"\n2.Pop"
				<<"\n3.Display"
				<<"\n4.Quit";
			cout<<"\nEnter the option : ";
			cin>>c;
			switch(c)
				{
```

```
                case 1:     cout<<"Enter value to be pushed : ";
                            cin>>v;
                            s.push(v);
                            break;
                case 2:     int i;
                            i=s.pop();
                            if(i!=-1)
                                  cout<<"Value popped is : "<<i<<endl;
                            break;
                case 3:     s.display();
                            break;
                case 4:     exit(0);
                            break;          }
        }while(1);                          }
```

```
_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 2
|Underflow|

_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 1
Enter value to be pushed : 34

_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 1
Enter value to be pushed : 23

_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 3
|Stack is as follows|
23      34
```

```
_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 2
Value popped is : 23

_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 2
Value popped is : 34

_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 3
|Empty stack|

_MENU_
1.Push
2.Pop
3.Display
4.Quit
Enter the option : 4
```

```
--------------------------------
Process exited after 41.66 seconds with return value 0
Press any key to continue . . .
```

**Q-2**
**Write a menu driven program from static implementation of QUEUE.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program to add, delete and display in queue.
#include<iostream>
#include<process.h>
using namespace std;
#define size 5
class queue
	{		int a[size];		int front,rear;
		public:
			queue()
				{	front=0;rear=0;  }
			void add(int);		int del();		void display();		};
void queue::add(int item)
	{	if(rear==size)		cout<<"|Queue is full|\n";
		else			a[rear++]=item;			}
int queue::del()
	{	if(front==rear)
			{	cout<<"|Queue is empty|\n";
				return -1;				}
		else		return a[front++];			}
void queue::display()
	{	int i;
		if(front==rear)
			{	cout<<"|Empty queue|\n";
				return;				}
		cout<<"|Queue is as follows|\n";
		for(i=front;i<rear;++i)		cout<<a[i]<<"\t";
		cout<<endl;					}
int main()
	{	queue q;
		int c,v;
		do
		{	cout<<"\n_MENU_\n";
			cout<<"1.Add"
				<<"\n2.Delete"
				<<"\n3.Display"
				<<"\n4.Quit";
			cout<<"\nEnter the option : ";
			cin>>c;
			switch(c)
				{
				case 1:	cout<<"Enter value to be added : ";
					cin>>v;
					q.add(v);
					break;
				case 2:	int i;
					i=q.del();
					if(i!=-1)
```

```
                        cout<<"Value deleted is : "<<i<<endl;
                        break;
                case 3: q.display();
                        break;
                case 4: exit(0);
                        break;          }
        }while(1);                      }
```

```
_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 2
|Queue is empty|

_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 1
Enter value to be added : 13

_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 1
Enter value to be added : 24

_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 3
|Queue is as follows|
13      24
```

```
_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 2
Value deleted is : 13

_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 2
Value deleted is : 24

_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 3
|Empty queue|

_MENU_
1.Add
2.Delete
3.Display
4.Quit
Enter the option : 4
```

```
--------------------------------
Process exited after 15.71 seconds with return value 0
Press any key to continue . . .
```

**Q-3**
**Write a menu driven program from static implementation of CIRCULAR QUEUE.**

**|PROGRAMMING and OUTPUT|**

```
//DEV C++
//Program for insertion,seletion and display in circular queue.
#include<iostream>
#include<process.h>
using namespace std;
#define size 5
class c_queue
      {           int a[size];        int front,rear;
            public:
                  c_queue()
                        {     front=rear=-1;   }
                  void insertion(int);      void deletion();   void display();    };
void c_queue::insertion(int item)
      {     if((front==0&&rear==size-1)||(front==rear+1))
                  cout<<"|Circular Queue is full|\n";
            else if(front==-1)
                  {     front=rear=0;
                        a[rear]=item;     }
            else if(rear==size-1)
                  {     rear=0;
                        a[rear]=item;     }
            else
                  {     ++rear;
                        a[rear]=item;     }                  }
void c_queue::deletion()
      {     if(front==-1)
                  {     cout<<"|Circular Queue is empty|\n";
                        return;           }
            else if(front==rear)    front=rear=-1;
            else if(front==size-1)  front=0;
            else  front++;                                  }
void c_queue::display()
      {     int i;
            if(front==-1)
                  {     cout<<"|Empty circular queue|\n";
                        return;           }
            cout<<"|Circular Queue is as follows|\n";
            if (rear>=front)
               {  for (int i=front;i<=rear;++i)       cout<<a[i]<<"\t";      }
            else
               {  for (int i=front;i<size;++i)        cout<<a[i]<<"\t";
                  for (int i=0;i<=rear;i++)           cout<<a[i]<<"\t";      }
            cout<<endl;                   }
int  main()
      {     c_queue cq;       int c,v;
            do
            {
            cout<<"\n_MENU_\n";
```

```
        cout<<"1.Insertion"<<"\n2.Deletion"<<"\n3.Display"<<"\n4.Quit";
        cout<<"\nEnter the option : ";      cin>>c;
        switch(c)
              {
              case 1:     cout<<"Enter value to be added : ";
                          cin>>v;
                          cq.insertion(v);
                          break;
              case 2:     int i;
                          cq.deletion();
                          break;
              case 3:     cq.display();
                          break;
              case 4:     exit(0);
                          break;
              }
        }while(1);                          }
```

```
_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 2
|Circular Queue is empty|

_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 1
Enter value to be added : 32

_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 1
Enter value to be added : 22

_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 3
|Circular Queue is as follows|
32      22
```

```
_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 2

_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 2

_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 3
|Empty circular queue|

_MENU_
1.Insertion
2.Deletion
3.Display
4.Quit
Enter the option : 4
```

```
--------------------------------
Process exited after 41.66 seconds with return value 0
Press any key to continue . . .
```