Phase -7

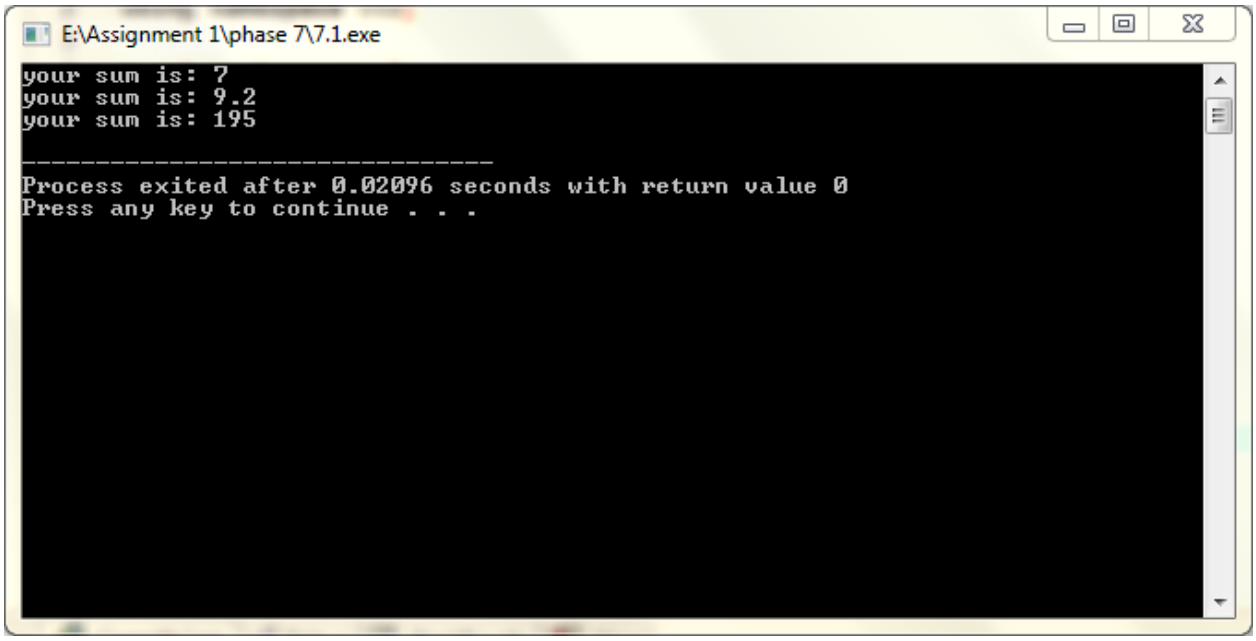
Practical -1

Aim 1:- Badal and Nayan wants to get out from a well located in a village farm. Another person stood outside at a well wants to help them. But near that well, two small pieces of ropes available which just not make any sense. By joining both that ropes, they will be able to get outside. So help that person to perform joining of that two pieces of certain length using concept of functional templates in C++.

Program:-

```
#include<iostream>
using namespace std;

template<typename T>
void add(T a, T b)
{
	cout<<"your sum is: "<<a+b<<endl;
}

int main()
{
	add<int>(5,2);
	add<float>(6.7f,2.5f);
	add<char>('a','b');
	return 0;
}
```

Output:


E:\Assignment 1\phase 7\7.1.exe
your sum is: 7
your sum is: 9.2
your sum is: 195

Process exited after 0.02096 seconds with return value 0
Press any key to continue . . .

Practical -2

Aim 2:- Design a Menu Driven Simple Calculator with a help of Class Templates. User should have a choice of following operations.
\ Additio
\ Subtraction
\ Multiplication
\ Division
\ Modulur
\ Exit

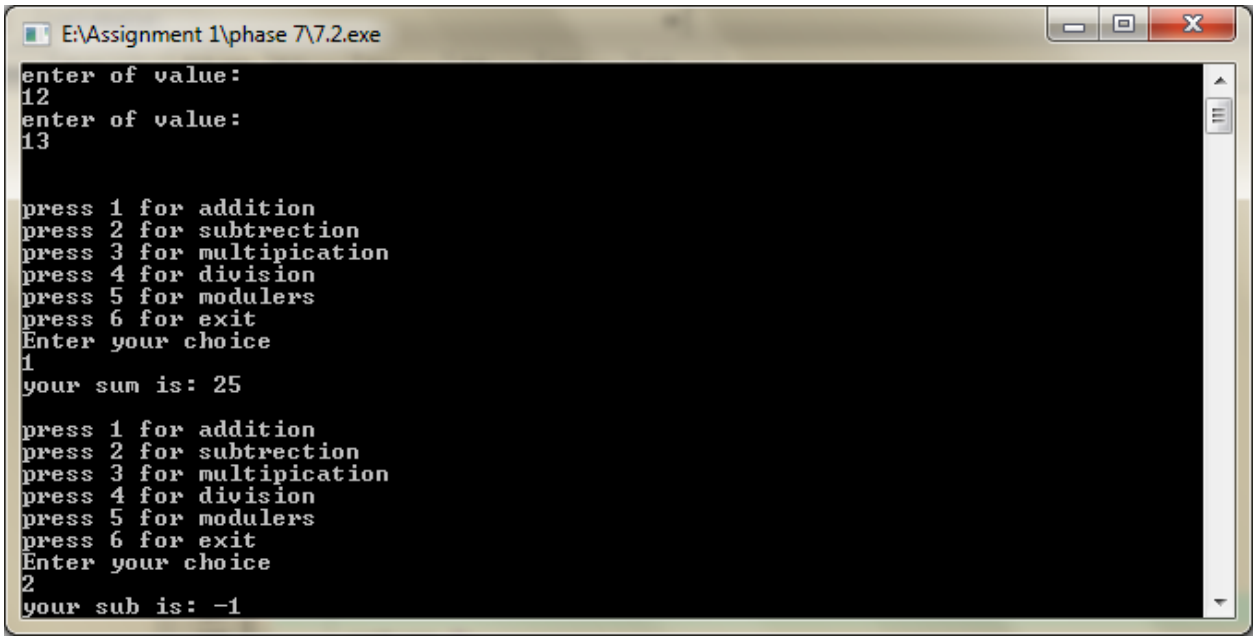
Program:-

```
#include<iostream>
using namespace std;

template<typename T>
class  Calculator
{
 public:

   add(T a,T b)
   {
         cout<<"your sum is: "<<a+b;
         }
         sub(T a,T b)
         {
            cout<<"your sub is: "<<a-b<<endl;
   }
   multi(T a,T b)
   {
         cout<<"your multi is: "<<a*b<<endl;
         }
         div(T a,T b)
         {
                  cout<<"your div is: "<<a/b<<endl;
         }
```

```
mod(T a,T b)
{
          cout<<"your mod is: "<<a%b<<endl;
}

int setdata()
{
                    int m,n,i=0,c;

                    cout<<"enter of value: "<<endl;
cin>>m;


cout<<"enter of value: "<<endl;
cin>>n;

 for(i=0;i<=1;i--)
 {
                     cout<<endl<<endl<<"press 1 for addition"<<endl;
                     cout<<"press 2 for subtrection"<<endl;
                     cout<<"press 3 for multipication"<<endl;
                     cout<<"press 4 for division"<<endl;
                     cout<<"press 5 for modulers"<<endl;
        cout<<"press 6 for exit"<<endl;


         cout<<"Enter your choice"<<endl;
         cin>>c;

          switch(c)
          {

             case 1:  add(m,n);
                 break;

             case 2:  sub(m,n);
                 break;


                         case 3:  multi(m,n);
```

```
                    break;


                    case 4: div(m,n);
                     break;


                    case 5: mod(m,n);
                break;


                    case 6:
                        return 0;
                     break;

              default:
                       cout<<"invild value"<<endl;

          }


       }

      }
};

int main()
{
      Calculator <int>c1;


      c1.setdata();

      return 0;

}
```

Output:


E:\Assignment 1\phase 7\7.2.exe
enter of value:
12
enter of value:
13
press 1 for addition
press 2 for subtrection
press 3 for multipication
press 4 for division
press 5 for modulers
press 6 for exit
Enter your choice
1
your sum is: 25
press 1 for addition
press 2 for subtrection
press 3 for multipication
press 4 for division
press 5 for modulers
press 6 for exit
Enter your choice
2
your sub is: -1

Practical -3

Aim 3:- Build a class template called "List" which accepts two parameters: Array & Size of that array. Define two List which have different elements of different size. Use both of that array to concat with each other and create another List. Now from that List, perform following operationsQ
\ Average value of that List
\ Max and Min value
Provide a Menu driven approach to do so.

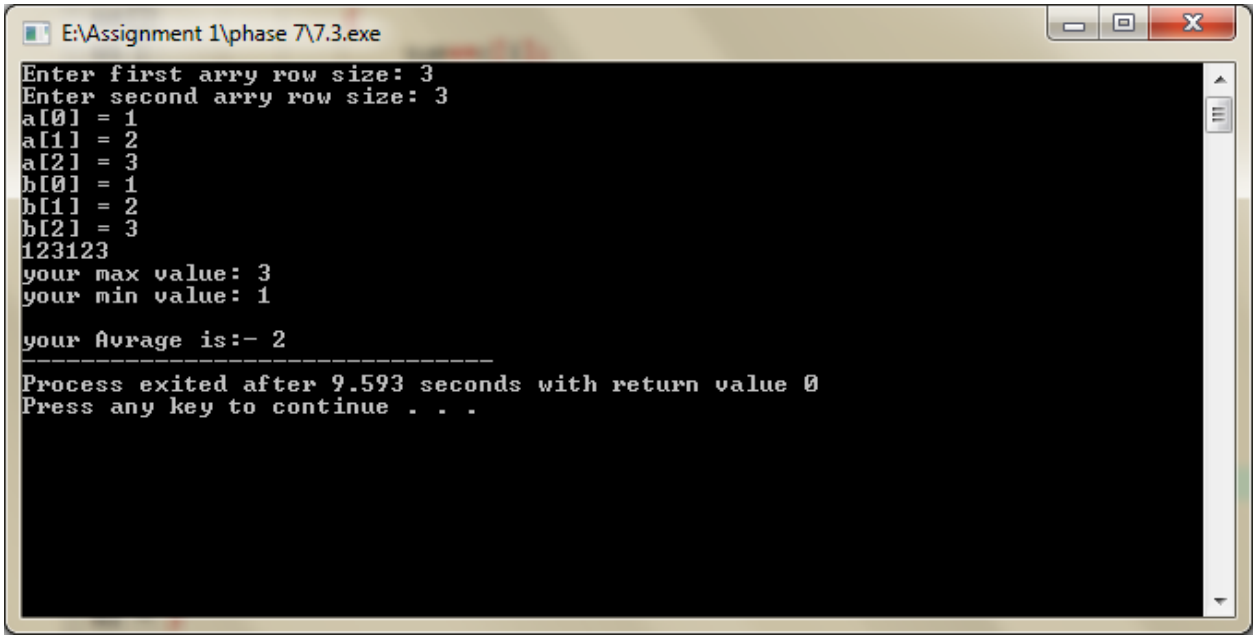
Program:-

```
#include<iostream>
using namespace std;

template<typename T>
class Avrage
{
        T a[100],b[100],c[100];
        int n,m,l,i,j;
        float sum=0;
        public:


                void setdata()
           {
             cout<<"Enter first arry row size: ";
                cin>>n;
                cout<<"Enter second arry row size: ";
                cin>>m;


                for(i=0;i<n;i++)
                {
                  cout<<"a["<<i<<"] = ";
                        cin>>a[i];
                }
```

```
  for(i=0;i<m;i++)
  {
    cout<<"b["<<i<<"] = ";
          cin>>b[i];
  }
}
void getdata()
{
   for(i=0;i<n;i++)
   {
         c[i]=a[i];
   }
   for(i=n,j=0; i<n+m,j<m;i++,j++)
   {
         c[i]=b[j];
         }
   for(i=0; i<n+m;i++)
   {
         cout<<c[i];
         }
         l=n+m;
         for(i=0; i<l;i++)
   {
         if(c[0]<=c[i])
         {
                  c[0]=c[i];
                  }
         }
         cout<<endl<<"your max value: "<<c[0]<<endl;
         for(i=0; i<l;i++)
   {
         if(c[0]>=c[i])
         {
                  c[0]=c[i];
                  }
         }
         cout<<"your min value: "<<c[0]<<endl;
  for(i=0; i<l;i++)
   {
         sum+=c[i];
```

```
                    }
                    cout<<endl<<"your Avrage is:- "<<sum/l;


                }


};
int main()
{
        Avrage <int>a1;
        a1.setdata();
        a1.getdata();


        return 0;
}
```

Output:



Practical -4

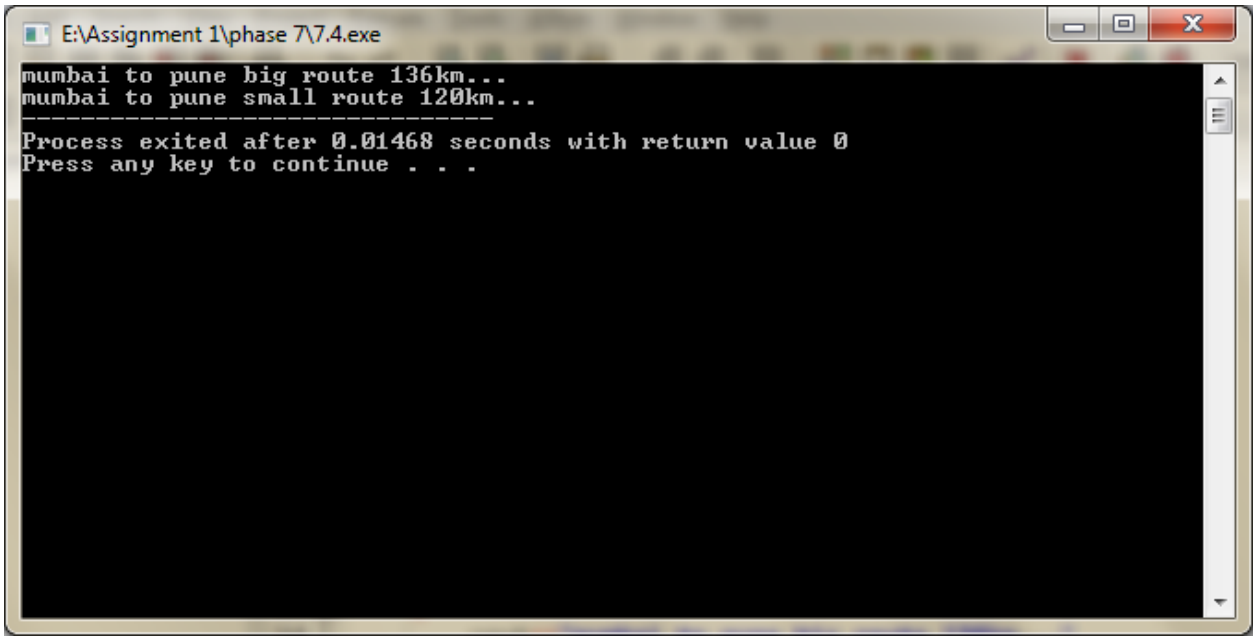
Aim 4:- Disha wants to help a Fire Brigade team by deciding Which of the following route is shorter to reach an Incident location:
Route A:mumbai to Pune -120KM
Route B:mumbai to Pune-136KM
Help Disha by designing function template to figure out Shorter route for Fire Brigade team.

Program:-

```
#include<iostream>
using namespace std;

template<typename T>
void m_to_p(T a,T b)
{
        if(a>b)
        {
                cout<<"mumbai to pune big route 120km..."
                   <<endl<<"mumbai to pune small route 136km...";
        }
        else
        {
                cout<<"mumbai to pune big route 136km..."
                   <<endl<<"mumbai to pune small route 120km...";
        }

}

int main()
{
        m_to_p(120,136);

        return 0;
```

}

Output:


E:\Assignment 1\phase 7\7.4.exe
mumbai to pune big route 136km...
mumbai to pune small route 120km...

Process exited after 0.01468 seconds with return value 0
Press any key to continue . . .

Practical -5

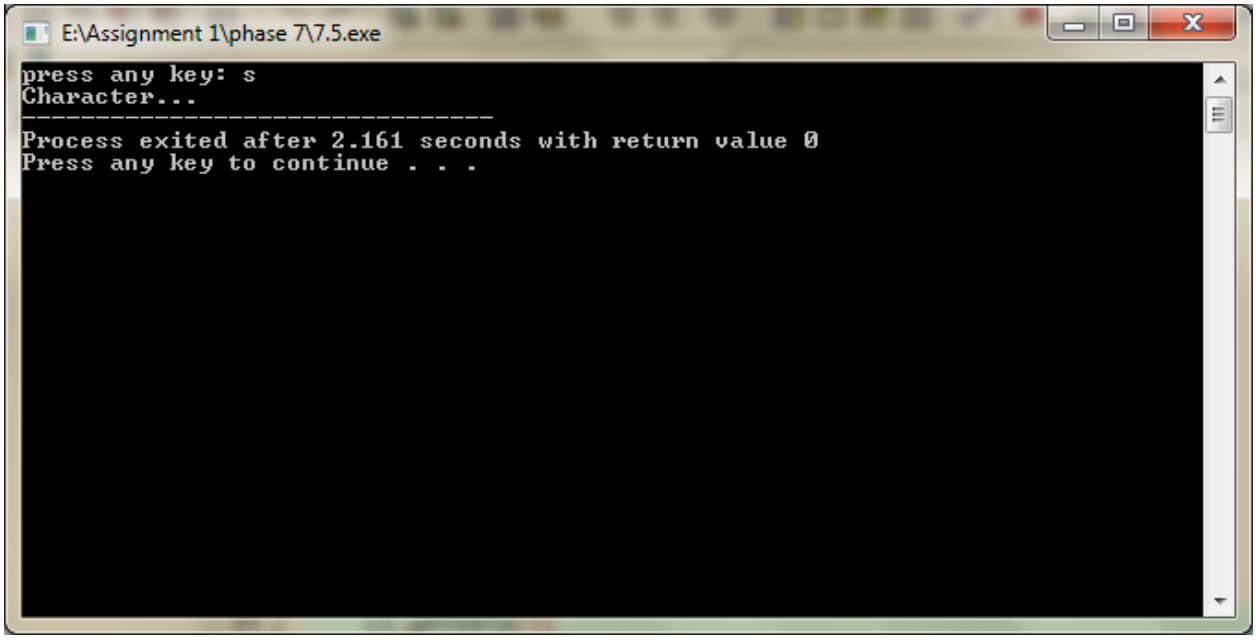
Aim 5:- Design an identification system for all letters Available from Keyboard.
For example.
If a user enters "B", then print"alphabet"
If a user enters "7", then print"digit"
If a user enters "#", then print"symbol"
To do so, use function/class template in C++.

Program:-

```
#include<iostream>
using namespace std;

template<typename T>
class Check
{
 public:
        T n;
        void setdata()
        {
                cout<<"press any key: ";
      cin>>n;
        }
        void getdata()
        {
                if(n>='0' &&  n<='9')
                {
                        cout<<"Digit...";
                }
                else if(n>='a' && n<='z' || n>='A' && n<='Z')
                {
                        cout<<"Character...";
                }
                else
                {
```

```
                cout<<"Special symbol...";
            }
        };

};

int main()
{
        Check <char>c1;

    c1.setdata();
    c1.getdata();


        return 0;
}
```

Output:



Practical -6

Aim 6:- SDJ college is organizing a competition for creating a Password strength detection system. Criterias for this System are As following:
Password must be greater than or equal to 6
Password must contains at least one alphabatic letter,
One digit and one special symbol.
So as a participant, build this type of system using class/function template in C++.

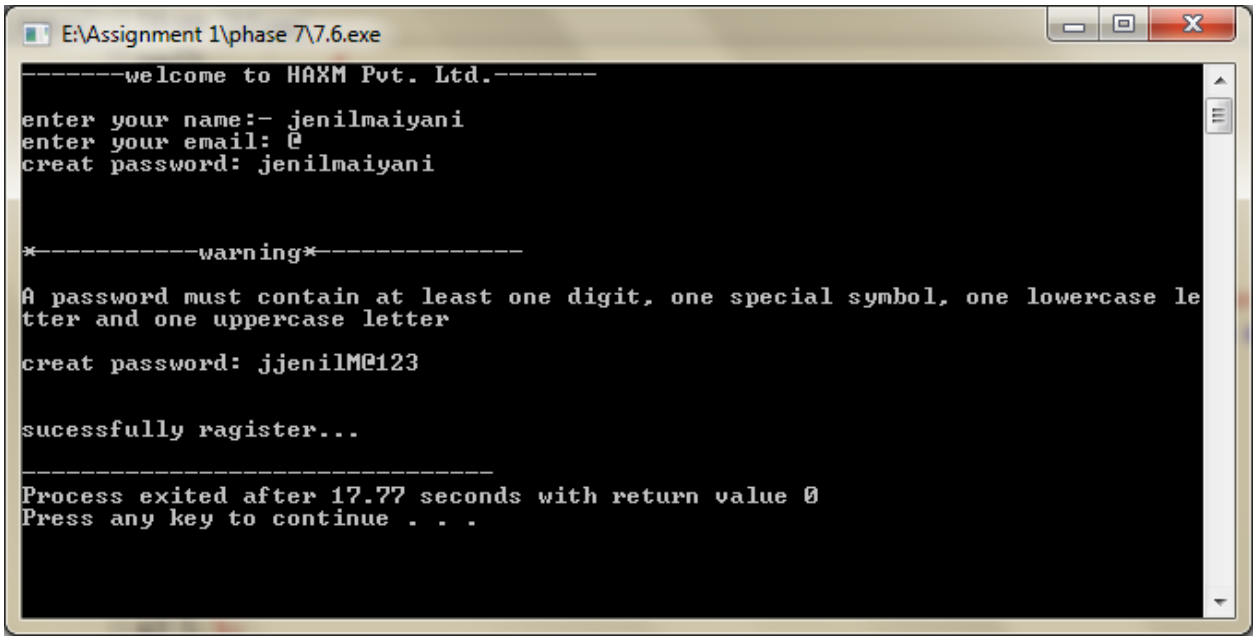
Program:-

```
#include<iostream>
#include<string.h>
using namespace std;

template<typename T>
class Account
{
public:
        T email[100],name[100];
        T password[100];
        int i,a=0,b=0,c=0,d=0;

        void setdata()
        {
        cout<<"-------welcome to HAXM Pvt. Ltd.-------"<<endl<<endl;

        cout<<"enter your name:- ";
        gets(name);

           cout<<"enter your email: ";
           gets(email);

   for(i=0; i<=1 ; i--)
        {
           cout<<"creat password: ";
                gets(password);
```

```
        cout<<endl<<endl;


        for(int i=0; password[i]!='\0';i++)
        {
            if(password[i]>='A' && password[i]<='Z')
        {
            a++;
            }
            else if(password[i]>='a' && password[i]<='z')
            {
                b++;
            }
            else if(password[i]>='0' && password[i]<='9')
            {
                c++;
            }
            else
            {
                d++;
            }
        }
        if(a>=1 && b>=1 && c>=1 && d>=1)
        {
                cout<<"sucessfully ragister..."<<endl;
                break;

        }
        else
        {
              cout<<endl<<"*-----------warning*--------------"<<endl<<endl;
              cout<<"A password must contain at least one digit, one special symbol,
one lowercase letter and one uppercase letter"<<endl<<endl;
        }
   }

      }
```

```
};
int main()
{
	Account <char>a1;
	a1.setdata();

	return 0;
}
```

Output:



Practical -7

Aim 7:- Design a "Find & Replace character(s) mechanism for Any given message or string. Use Non-type parameters for template.

Program:-

```
#include<iostream>
#include<string.h>
using namespace std;
template<typename T, int j>
class replace
{
   private:
      T s[100];
      T w,n;

   public:
      replace()
      {
         cout<<"Enter the String : ";
         gets(this->s);

         cout<<"which word you need to replace : ";
         cin>>w;

         cout<<"What is new word : ";
         cin>>n;

         for(int i=0; s[i]!='\0';i++)
         {
                  if(s[i]==w)
                  {
                           s[i] = n;
                                    }

                           }
```

```
    }
    void getdata()
    {
        cout<<"your updated String is : "<<s;
    }

};
int main()
{
   replace <char, 5>o1;
   o1.getdata();
   return 0;
}
```

Output: