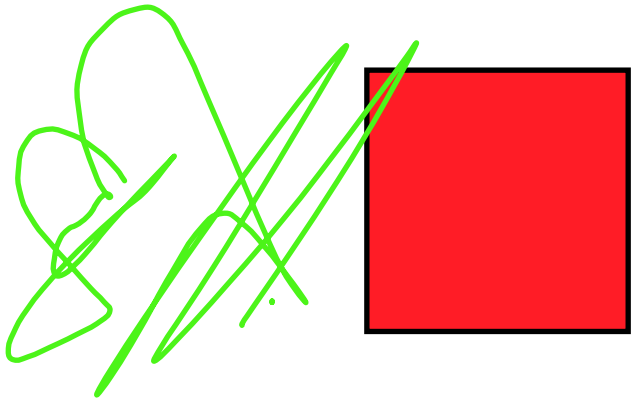
lab 2.1



manoj

```
#include <iostream>

using namespace std;

class swap2;

class swap1

{

private:

   int a;

   friend void swap(swap1,swap2);

public:

   swap1(int n){a=n;}

};

class swap2

{

private:

   int b;

   friend void swap(swap1,swap2);

public:

   swap2(int n){b=n;}

};

void swap(swap1 s1,swap2 s2)

{

   cout<<"before swapping the value of a and b is::"<<s1.a<<" "<<s2.b<<endl;

   int temp;

   temp=s1.a;
```

```
   s1.a=s2.b;
   s2.b=temp;
   cout<<"after swapping the value of a and b is ::"<<s1.a<< " "<<s2.b;
}

int main()
{
  swap1 s1(4);
  swap2 s2(5);
  swap(s1,s2);
   return 0;
}
```

Lab 2.2

```
#include <iostream>

using namespace std;
class us_currency;
class nepali_currency
{
   float rupee;
public:
   friend int operator>(nepali_currency,us_currency);
   friend int operator<(nepali_currency,us_currency);
   friend int operator==(nepali_currency,us_currency);
   void getdata()
   {
```

```
        cout<<"enter money in rupee\n";
        cin>>rupee;
    }
};
class us_currency
{
    float dollar;
public:
    friend int operator >(nepali_currency,us_currency);
    friend int operator<(nepali_currency,us_currency);
    friend int operator==(nepali_currency,us_currency);
    void getin()
    {
        cout<<"enter money in dollar";
        cin>>dollar;
    }
};
int operator==(nepali_currency n1,us_currency u1)
{
    u1.dollar=u1.dollar*101.36;
    if(u1.dollar==n1.rupee)
        return 1;
        else
        return 0;
}
```

```
int operator>(nepali_currency n1,us_currency u1)
{
    u1.dollar=u1.dollar*101.36;

   if(u1.dollar<n1.rupee)

      return 1;
      else
         return 0;

}
int operator<(nepali_currency n1,us_currency u1)
{
    u1.dollar=u1.dollar*101.36;
    if(u1.dollar>n1.rupee)
      return 1;
      else
         return 0;
}



int main()
{
   nepali_currency n1;
```

```
   us_currency u1;
   n1.getdata();
   u1.getin();
  if(n1>u1)
   cout<<"nepali rupee is greater than us dollar";
 else if(n1<u1)
      cout<<"\nUS dollar is greater than nepali rupee ";
   else if(n1==u1)
   cout<<"both currencies are equal";

   return 0;
}
```

****

lab 2.2.1 (using friend class according our dai)

```
#include <iostream>

using namespace std;
class  nepali_currency
{
   float rupee;
public:
   friend class us_currency;

   void getdata()
   {
      cout<<"enter the rupee\n";
```

```
        cin>>rupee;
    }
};
class us_currency
{
    float dollar;
public:

    void getdata()
    {
        cout<<"enter the dollar\n";
        cin>>dollar;
    }
    int operator>(nepali_currency a1)
    {
        dollar=dollar*101.36;
        if(dollar>a1.rupee)
            return 1;
        else
            return 0;
    }
    int operator<(nepali_currency a1)
    {
        dollar=dollar*101.36;
        if(dollar<a1.rupee)
```

```
        return 1;
      else
        return 0;
   }
   int operator ==(nepali_currency a1)
   {
      dollar=dollar*101.36;
      if(dollar==a1.rupee)
        return 1;

   }
};

int main()
{
  nepali_currency a;
  us_currency b;
  a.getdata();
  b.getdata();
   if(b>a)
      cout<<"us dollar is greater than nepali rupee\n";
   else if(b<a)
      cout<<"nepali rupee is greater than us dollar\n";
   else if(b==a)
      cout<<"both currencies are equal";
```

```
   return 0;
}
```

****

lab 2.3**

```
#include <iostream>
#include<math.h>
using namespace std;
class complex
{
   float rl,img;
public:
   void getdata()
   {
      cin>>rl>>img;
   }
   void showdata()
   {
      if(img>0)
      cout<<rl<<"+"<<img<<"i\n";
      else
      cout<<rl<<img<<"i\n";

   }
   void operator +(complex c2)
   {
```

```
  complex temp;
  temp.rl=rl+c2.rl;
  temp.img=img+c2.img;
  cout<<"after addition result is::";
  temp.showdata();
}
void operator -(complex c2)
{
   complex temp;
  temp.rl=rl-c2.rl;
  temp.img=img-c2.img;
  cout<<"after subtraction result is::";
  temp.showdata();
}
void operator*(complex c2)
{
   complex temp;
  temp.rl=(rl*c2.rl-img*c2.img);
  temp.img=(rl*c2.img+img*c2.rl);
  cout<<"after multiplication result is::";
  temp.showdata();
}
void operator/(complex c2)
{
   complex temp;
```

```
        temp.rl=(rl*c2.rl+img*c2.img)/(pow(c2.rl,2)+pow(c2.img,2));
       temp.img=(img*c2.rl-rl*c2.img)/(pow(c2.rl,2)+pow(c2.img,2));
       cout<<"after division result is::";
       temp.showdata();


    }


};


int main()
{
    complex c1,c2,c3;
    cout<<"enter the first real number and imaginary \n";
    c1.getdata();
    cout<<"enter the second real number and imaginary \n";
    c2.getdata();
    c1+c2;
    c1-c2;
    c1*c2;
    c1/c2;
    return 0;
}
```

Lab 2.4**

```
#include <iostream>


using namespace std;
```

```
class time
{
   int hr,min,sec;
public:
   time(){hr=0;min=0;sec=0;}
   time(int h,int m,int s)
   {
      hr=h;
      min=m;
      sec=s;
   }
   void showdata()
   {
      cout<<hr<<" "<<min<<" "<<sec<<endl;
   }
   void operator +(time t2)
   {
      time temp;
      temp.hr=hr+t2.hr;
      temp.min=min+t2.min;
      temp.sec=sec+t2.sec;
      if(temp.sec>=60){
      int a=temp.sec/60;
         temp.min=temp.min+a;
         temp.sec=temp.sec%60;
```

```
    }
    else if(temp.min>=60)
    {
       int b=temp.min/60;
       temp.hr=temp.hr+b;
       temp.min=temp.min%60;
    }
    cout<<"after adding result is::\n";
    temp.showdata();
}
void operator -(time t2)
{
    time temp;
    temp.hr=hr-t2.hr;
    temp.min=min-t2.min;
    temp.sec=sec-t2.sec;
    cout<<"after subtraction the result is::\n";
    temp.showdata();
}
void operator >(time t2)
{
    time temp;

    if(hr>t2.hr)
    {
```

```
        temp.hr=hr-t2.hr;
        temp.min=min-t2.min;
        temp.sec=sec-t2.sec;
        cout<<"time t1 is greater than time t2 by:::\n";
        temp.showdata();
      }
      else
      {
        cout<<"time t1 is less than time t2 by:::\n";
        temp.showdata();
      }

  }
  time operator<(time t2)
  {

  }

};

int main()
{
  cout<<"**************************ALL RESULT IN
HH::MM::SS*****************************************\n";
  time t1(7,20,55),t2(5,10,50),t3;
  t1+t2;
```

```
   t1-t2;
   t1>t2;
   return 0;
}
```

lab 2.5

```
#include <iostream>
#include<cstring>
using namespace std;
class STRING
{
   char *s;
   int l;
public:
   STRING (){s=new char('\0');}
   STRING(char s1[])
   {
     l=strlen(s1);
     s=new char[l+1];
     strcpy(s,s1);
   }
   void display()
   {
      cout<<s<<endl;
   }
   STRING operator+(STRING s2)
   {
```

```
    STRING t1;
    t1.s=new char[l+s2.l];
    strcpy(t1.s,s);
    strcat(t1.s,s2.s);
    return t1;
  }
  void operator =(STRING s1)
  {
    STRING t1;
    t1.s=new char[s1.l];
    strcpy(t1.s,s1.s);
    t1.display();

  }
  int operator==(STRING s1)
  {
    if(strcmp(s,s1.s)==0)
      return 1;
    else
      return 0;
  }

};
```

```
int main()
{
   STRING s1("Hi");
   STRING s2(" my name is manoj nepali");
   STRING s3;
   s3=s1+s2;
   s3.display();
   s2=s1;
   if(s2==s1)
      cout<<"both strings are same\n";
   else
      cout<<"both the strings are different\n";



   return 0;
}
**Lab 2.6**
#include <iostream>
#include<cstring>

using namespace std;
class STRING
{
private:
   char str[100];
```

```
  int l;
public:
  STRING(){}
  STRING(char s[])
  {
    strcpy(str,s);
    l=strlen(str);
  }
  STRING operator+(STRING s2)
  {
    STRING t2;
    if(l+s2.l<100)
    {
      strcpy(t2.str,str);
      strcat(t2.str,s2.str);
      return t2;
    }
    else
      cout<<"insufficient space\n";

  }
   void display()
   {
     cout<<str<<endl;
   }
```

```
    void operator=(STRING s1)
    {
       STRING t2;
       strcpy(t2.str,s1.str);
             t2.display();

    }
    int operator==(STRING s1)
    {

       if(strcmp(s1.str,str)==0)
         return 1;
         else
            return 0;
    }
};

int main()
{
   STRING s1("Hi");
   STRING s2(" my name is manoj nepali");
   STRING s3,s4;

   s3=s1+s2;
   s3.display();
```

```
   s2=s1;
   if(s2==s1)
      cout<<"both are equal\n";
   else
      cout<<"both string are different";

   return 0;
}
```

lab 2.7

```
#include <iostream>

using namespace std;
class time
{
   int hr,min,sec;
public:
   friend void operator>>(istream&,time&);
    friend void operator<<(ostream&,time);
};
void operator>>(istream &in,time &t1)
{
   in>>t1.hr>>t1.min>>t1.sec;
}
void operator<<(ostream &out,time t1)
{
```

```
   out<<t1.hr<<":"<<t1.min<<":"<<t1.sec;
}



int main()
{
   time t1;
   cout<<"enter time in hr min and sec\n";
   cin>>t1;
   cout<<"the time is\n";
   cout<<t1;


   return 0;
}
**lab2.9**
#include <iostream>
#include<stdlib.h>

using namespace std;
class matrix
{
   int m,n;
   int a[20][20];
public:
   void get_rowcolumn()
```

```
{
   cin>>m>>n;
}
void getmatrix()
{
   for(int i=0;i<m;i++)
   {
      for(int j=0;j<n;j++)
         cin>>a[i][j];
   }
}
void displaymatrix()
{
   cout<<"the result is::::\n";
   for(int i=0;i<m;i++)
   {
      for(int j=0;j<n;j++){

         cout<<a[i][j]<<" ";
         }
      cout<<"\n";
   }

}
matrix operator +(matrix m2)
```

```
    {
        matrix temp;
        temp.m=m;
        temp.n=n;
         for(int i=0;i<m;i++)
        {
            for(int j=0;j<m2.n;j++)
              temp.a[i][j]=a[i][j]+m2.a[i][j];
    }
    return temp;
    }

};

int main()
{
    matrix m1,m2,m3;
    cout<<"enter the first row and column\n";
    m1.get_rowcolumn();
    cout<<"enter the second row and column\n";
    m2.get_rowcolumn();
    cout<<"enter the first matrix\n";
    m1.getmatrix();
    cout<<"enter the second matrix\n";
    m2.getmatrix();
```

```
    m3=m1+m2;
    m3.displaymatrix();
    return 0;
}
```

**********FINISH**********