Name : Laukik Bhagawan Bhogale
Roll no. 14
Batch A

Program Assignment no 01

```
#include<math.h>
#include<iostream>
using namespace std;
class SEIT
{
int n;
 struct Student{
    int rn;
    string name;
    float sgpa;
 }
 s[20];
    public:
    void setdata();
    void getdata();
    void search(float);
    void boubble_sort();
    void insertion();
    void binary_search(string key);
    void quick_sort(int,int);
    int partition(int, int);
    void quick_display(int);
};

 void SEIT::setdata()
  {
        cout<<"Enter number of students"<<endl;
        cin>>n;
        for(int i=0;i<n;i++)
          {
                cout<<"Enter roll no,name,sgpa"<<endl;
                cin>>s[i].rn>>s[i].name>>s[i].sgpa;
          }
  }

 void SEIT::getdata()
  {
        cout<<"Roll no:"<<"\t"<<"name: "<<"\t"<<"\t"<<"SGPA: "<<endl;
        for(int i=0;i<n;i++)
         {
                cout<<s[i].rn<<"\t"<<"\t"<<s[i].name<<"\t"<<"\t"<<s[i].sgpa<<endl;
         }
  }


 void SEIT::search(float marks)
  {
    cout<<"Roll no:"<<"\t"<<"name:"<<"\t"<<"\t"<<"SGPA:"<<endl;
```

```
        for(int i=0;i<n;i++)
        {
                if(s[i].sgpa>=marks)
                {
                        cout<<s[i].rn<<"\t"<<"\t"<<s[i].name<<"\t"<<"\t"<<s[i].sgpa<<endl;
                }
        }
  }

void SEIT::boubble_sort()
  {
        for(int i=0;i<n;i++)
        {
                for(int j=0;j<n-i-1;j++)
                {
                        if(s[j].rn>s[j+1].rn)
                        {
                                struct Student temp=s[j];
                                s[j]=s[j+1];
                                s[j+1]=temp;
                        }
                }
        }
        getdata();
  }

 void SEIT::insertion()
  {
        int i,p;
        struct Student temp;
        for(i=1;i<n;i++)
        {
                string key=s[i].name;
                temp=s[i];
                for(p=i-1;(p>=0) && s[p].name>key;p--)
                {
                        s[p+1]=s[p];
                }
                s[p+1]=temp;
        }
        getdata();
  }

void SEIT::binary_search(string key)
  {
        int l=0,h=n-1,mid;
        while(l<=h)
        {
                mid=floor(l+h)/2;
                if(key==s[mid].name)
                {
                        cout<<"Key Found: "<<key;
                        break;
                }
                else if(key<s[mid].name)
                {
                        h=mid-1;
```

```
                }
                else if(key>s[mid].name)
                {
                        l=mid+1;
                }
                else
                {
                         cout<<"Key not found";
                         break;
                }
        }

  }

int SEIT::partition(int l,int h)
 {
        int i,j;
        struct Student pvt,tem;
        pvt=s[l];
        i=l+1;
        j=h;

        while(i<=j)
        {
                while(s[i].sgpa<=pvt.sgpa)
                {
                        i++;
                }
                while(s[j].sgpa>pvt.sgpa)
                {
                        j--;
                }
                if(i<j)
                {
                        tem=s[i];
                        s[i]=s[j];
                        s[j]=tem;
                }
        }
        s[l]=s[j];
        s[j]=pvt;
        return j;
 }

void SEIT::quick_sort(int l,int h)
{
        int j;
        if(l<h)
        {
                j=partition(l,h);
                quick_sort(l,j-1);
                quick_sort(j+1,h);
        }
}

 void SEIT::quick_display(int t)
 {#include<math.h>
```

```
#include<iostream>
using namespace std;
class SEIT
{
int n;
 struct Student{
    int rn;
    string name;
    float sgpa;
 }
 s[20];
    public:
    void setdata();
    void getdata();
    void search(float);
    void boubble_sort();
    void insertion();
    void binary_search(string key);
    void quick_sort(int,int);
    int partition(int, int);
    void quick_display(int);
};

  void SEIT::setdata()
   {
         cout<<"Enter number of students"<<endl;
         cin>>n;
         for(int i=0;i<n;i++)
           {
                cout<<"Enter roll no,name,sgpa"<<endl;
                cin>>s[i].rn>>s[i].name>>s[i].sgpa;
           }
   }

  void SEIT::getdata()
   {
         cout<<"Roll no:"<<"\t"<<"name: "<<"\t"<<"\t"<<"SGPA: "<<endl;
         for(int i=0;i<n;i++)
          {
                cout<<s[i].rn<<"\t"<<"\t"<<s[i].name<<"\t"<<"\t"<<s[i].sgpa<<endl;
          }
   }


  void SEIT::search(float marks)
   {
     cout<<"Roll no:"<<"\t"<<"name:"<<"\t"<<"\t"<<"SGPA:"<<endl;
         for(int i=0;i<n;i++)
         {
                if(s[i].sgpa>=marks)
                {
                        cout<<s[i].rn<<"\t"<<"\t"<<s[i].name<<"\t"<<"\t"<<s[i].sgpa<<endl;
                }
         }
   }

 void SEIT::boubble_sort()
```

```
 {
        for(int i=0;i<n;i++)
        {
                for(int j=0;j<n-i-1;j++)
                {
                        if(s[j].rn>s[j+1].rn)
                        {
                                struct Student temp=s[j];
                                s[j]=s[j+1];
                                s[j+1]=temp;
                        }
                }
        }
        getdata();
 }

 void SEIT::insertion()
 {
        int i,p;
        struct Student temp;
        for(i=1;i<n;i++)
        {
                string key=s[i].name;
                temp=s[i];
                for(p=i-1;(p>=0) && s[p].name>key;p--)
                {
                        s[p+1]=s[p];
                }
                s[p+1]=temp;
        }
        getdata();
 }

void SEIT::binary_search(string key)
 {
        int l=0,h=n-1,mid;
        while(l<=h)
        {
                mid=floor(l+h)/2;
                if(key==s[mid].name)
                {
                        cout<<"Key Found: "<<key;
                        break;
                }
                else if(key<s[mid].name)
                {
                        h=mid-1;
                }
                else if(key>s[mid].name)
                {
                        l=mid+1;
                }
                else
                {
                         cout<<"Key not found";
                         break;
                }
```

```
        }

  }

int SEIT::partition(int l,int h)
 {
        int i,j;
        struct Student pvt,tem;
        pvt=s[l];
        i=l+1;
        j=h;

        while(i<=j)
        {
                while(s[i].sgpa<=pvt.sgpa)
                {
                        i++;
                }
                while(s[j].sgpa>pvt.sgpa)
                {
                        j--;
                }
                if(i<j)
                {
                        tem=s[i];
                        s[i]=s[j];
                        s[j]=tem;
                }
        }
        s[l]=s[j];
        s[j]=pvt;
        return j;
 }

void SEIT::quick_sort(int l,int h)
{
        int j;
        if(l<h)
        {
                j=partition(l,h);
                quick_sort(l,j-1);
                quick_sort(j+1,h);
        }
}

 void SEIT::quick_display(int t)
 {
  int i;
 cout<<"Roll no:"<<"\t"<<"name:"<<"\t"<<"SGPA:"<<endl;
 for(i=n;i>n-t;i--)
 {
 cout<<s[i].rn<<"\t"<<s[i].name<<"\t"<<s[i].sgpa<<endl;
 }
}

int main()
 {
```

```
 SEIT p;

        while(1)
        {
                cout<<"\n  ****Menu****";
                cout<<"\n1.Input the Student data";
                cout<<"\n2.Information about all Student ";
                cout<<"\n3.Student Information whose SGPA is greater than 8";
                cout<<"\n4.Informatin according to Roll No. ";
                cout<<"\n5.Informatin according to Student Name ";
                cout<<"\n6.Check student is present or not ";
                cout<<"\n7.Top Student";
                cout<<"\n8.exit"<<endl;

                cout<<"\nEnter the Choice: ";
                int ch;
                cin>>ch;

                switch(ch)
                {
                        case 1:
                                 p.setdata();
                                 break;
                        case 2:
                                p.getdata();
                                break;
                        case 3:
                                p.search(8);
                                break;
                        case 4:
                                p.boubble_sort();
                                break;
                        case 5:
                                p.insertion();
                                break;
                        case 6:
                                p.binary_search("Laukik");
                                break;
                        case 7:
                                p.quick_sort(0,3);
                                p.quick_display(2);
                                break;
                        case 8:
                                exit(0);
                }
        }
}
```

OUTPUT :

```
 ****Menu****
1.Input the Student data
2.Information about all Student
```

3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 1
Enter number of students
3
Enter roll no,name,sgpa
1
pravin
9.1
Enter roll no,name,sgpa
2
rushikesh
7.9
Enter roll no,name,sgpa
3
Laukik
9.3

****Menu****
1.Input the Student data
2.Information about all Student
3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 2

Roll no:	name:	SGPA:
1	pravin	9.1
2	rushikesh	7.9
3	Laukik	9.3

****Menu****
1.Input the Student data
2.Information about all Student
3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 3

Roll no:	name:	SGPA:
1	pravin	9.1
3	Laukik	9.3

****Menu****
1.Input the Student data
2.Information about all Student

3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 4

Roll no:	name:	SGPA:
1	pravin	9.1
2	rushikesh	7.9
3	Laukik	9.3

****Menu****
1.Input the Student data
2.Information about all Student
3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 5

Roll no:	name:	SGPA:
3	Laukik	9.3
1	pravin	9.1
2	rushikesh	7.9

****Menu****
1.Input the Student data
2.Information about all Student
3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 6
Key Found: Laukik
****Menu****
1.Input the Student data
2.Information about all Student
3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 7

Roll no:	name:	SGPA:
3	Laukik	9.3
1	pravin	9.1

****Menu****

1.Input the Student data
2.Information about all Student
3.Student Information whose SGPA is greater than 8
4.Informatin according to Roll No.
5.Informatin according to Student Name
6.Check student is present or not
7.Top Student
8.exit

Enter the Choice: 8