ASSIGNMENT 1 – SORTING

```
#include <iostream>
#include <string.h>
#include <list>

using namespace std;

struct student
{
    int rollno;
    char name[20];
    float sgpa;
};

void accept(struct student list[5]);
void display(struct student list[80]);
void bubblesort(struct student list[5]);
void insertsort(struct student list[5]);
void search(struct student list[5]);
void binarysearch(struct student list[5]);

int main()
{
    struct student data[20];

    int ch;

    accept(data);

    do
    {
```

```
        cout << "Choices: 1.Bubble SORT\n2.Insertion
SORT\n3.SEARCH\n4.BINSEARCH" << endl;
        cin >> ch;

        switch (ch)
        {
        case 1:
            bubblesort(data);
            display(data);
            break;

        case 2:
            insertsort(data);
            display(data);
            break;

        case 3:
            search(data);
            break;

        case 4:
            binarysearch(data);
            break;
        }
    } while (ch != 4);

    return 0;
}

void accept(struct student list[5])
{
    for (int i = 0; i < 5; i++)
```

```
    {
        cout << "Enter roll no, name, sgpa: ";
        cin >> list[i].rollno >> list[i].name >> list[i].sgpa;
    }
}

void display(struct student list[80])
{
    cout << "Roll NO"
         << "\t"
         << "Name"
         << "\t"
         << "SGPA"
         << "\n";

    for (int i = 0; i < 5; i++)
    {
        cout << "\n"
             << list[i].rollno << "\t" << list[i].name << "\t" <<
list[i].sgpa << endl;
    }
}

void bubblesort(struct student list[5])
{
    struct student temp;

    int size = 5;

    for (int i = 0; i < size - 1; i++)
    {
        for (int j = 0; j < (size - 1 - i); j++)
```

```
        {
            if (list[j].rollno > list[j + 1].rollno)
            {
                temp = list[j + 1];
                list[j + 1] = list[j];
                list[j] = temp;
            }
        }
    }
}

void insertsort(struct student list[5])
{
    struct student temp;
    int k;
    int j;

    for (k = 0; k < 5; k++)
    {
        temp = list[k];
        j = k - 1;

        while (strcmp(list[j].name, temp.name) > 0 && j >= 0)
        {
            list[j + 1] = list[j];
            --j;
        }
        list[j + 1] = temp;
    }
}

void search(struct student list[5])
```

```
{
    float SGPA;
    int i;

    cout << "\nEnter SGPA: ";
    cin >> SGPA;

    cout << "\n\nRollno\tName\tSGPA\n";
    for (i = 0; i < 5; i++)
    {
        if (SGPA == list[i].sgpa)
        {
            cout << "\n"
                 << list[i].rollno << "\t" << list[i].name << "\t" <<
list[i].sgpa;
        }
    }
}

void binarysearch(struct student list[5])
{
    char search[80];
    int upper, lower, mid;
    int size = 5;

    cout << "Enter name: ";
    cin >> search;

    lower = 0;
    upper = size - 1;

    mid = (upper + lower) / 2;
```

```
    while (lower <= upper)
    {
        if (strcmp(list[mid].name, search) < 0)
        {
            lower = mid + 1;
        }
        else if (strcmp(list[mid].name, search) == 0)
        {
            cout << "\n"
                 << list[mid].rollno << "\t" << list[mid].name << "\t"
<< list[mid].sgpa;
            break;
        }
        else
        {
            upper = mid - 1;
        }
        mid = (upper + lower) / 2;
    }
}
```

OUTPUT:

```
Enter roll no, name, sgpa: 3 YASH 7.5
Enter roll no, name, sgpa: 1 ALICE 8.2
Enter roll no, name, sgpa: 5 BOB 6.9
Enter roll no, name, sgpa: 2 CHARLIE 7.8
Enter roll no, name, sgpa: 4 DAVE 7.0
Choices: 1.BSORT
2.ISORT
3.SEARCH
4.BINSEARCH
```

```
1
Roll NO Name    SGPA

1       ALICE   8.2

2       CHARLIE 7.8

3       YASH    7.5

4       DAVE    7

5       BOB     6.9
Choices: 1.BSORT
2.ISORT
3.SEARCH
4.BINSEARCH
1
Roll NO Name    SGPA

1       ALICE   8.2

2       CHARLIE 7.8

3       YASH    7.5

4       DAVE    7

5       BOB     6.9
Choices: 1.BSORT
2.ISORT
3.SEARCH
4.BINSEARCH
```

```
2
Roll NO Name    SGPA

1       ALICE   8.2

5       BOB     6.9

2       CHARLIE 7.8

4       DAVE    7

3       YASH    7.5
Choices: 1.BSORT
2.ISORT
3.SEARCH
4.BINSEARCH
3

Enter SGPA: 6.9


Rollno  Name    SGPA

5       BOB     6.9
Choices: 1.BSORT
2.ISORT
3.SEARCH
4.BINSEARCH
4
Enter name: BOB

5       BOB     6.9
```