**ASSIGNMENT 1 – Searching And Sorting**

**#bubble\_sort.h**

void BubbleSort(int arr[],const int size){

int tmp;

for(int i = 0 ; i < size ; i++ )

for(int j = i + 1 ; j < size ; j++)

if(arr[j] < arr[i]){

tmp = arr[j];

arr[j] = arr[i];

arr[i] = tmp;

}

}

**#selection\_sort.h**

void SelectionSort(int arr[] , const int size){

int min\_pos,tmp;

for(int i = 0 ; i < size ; i++ ){

min\_pos = i;

for(int j = i + 1 ; j < size ; j++)

if( arr[j] < arr[min\_pos])

min\_pos = j;

//Swapping minimum with arr[i]

tmp = arr[i];

arr[i] = arr[min\_pos];

arr[min\_pos] = tmp;

}

}

**#linear\_search.h**

int LinearSearch(const int arr[],const int size){

int x;

printf("Enter the number to search for: ");

scanf("%d",&x);

int rVal = -1; //Return Value

for(int i = 0 ; i < size ; i++)

if(arr[i] == x){

rVal = i;

break;

}

return rVal;

}

**#binary\_search.h**

extern void SelectionSort(int[],const int);

extern void Display(const int[],const int);

int BinarySearch(int arr[],const int size){

int x;

printf("Enter the number to search for: ");

scanf("%d",&x);

int rVal = -1; //Return Value

//Sorting before binary search

SelectionSort(arr,size);

Display(arr,size);

int left = 0 , right = size - 1,middle;

while(left<=right){

middle = (left + right)/2;

if(arr[middle] == x){

rVal = middle;

break;

}

else if(x>arr[middle])

left = middle + 1;

else

right = middle - 1;

}

return rVal;

}

**#ex1.c //Main Program**

#include <stdio.h>

#include "binary\_search.h"

#include "linear\_search.h"

#include "bubble\_sort.h"

#include "selection\_sort.h"

void Display(const int[],const int);

int main(void){

int arr[50],n,tmp,stop = 0,opt,index;

while(stop == 0){

printf("Enter the number of elements in the array: ");

scanf("%d", &n);

printf("Enter all the elements: ");

for(int i = 0 ; i < n ; i++)

scanf("%d",&arr[i]);

printf("1 - Selection Sort\n2 - Bubble Sort\n3 - Linear Search\n4 - Binary Search\nEnter your choice: ");

scanf("%d",&opt);

switch(opt){

case 1: SelectionSort(arr,n);

Display(arr,n);

break;

case 2: BubbleSort(arr,n);

Display(arr,n);

break;

case 3: index = LinearSearch(arr,n);

if(index != -1)

printf("Element found at index: %d\n",index);

else

printf("Element not found!\n");

break;

case 4: index = BinarySearch(arr,n);

if(index != -1)

printf("Element found at index: %d\n",index);

else

printf("Element not found!\n");

break;

default:printf("Invalid Input!Try again!\n");

}

printf("Do you want to stop 1/0: ");

scanf("%d",&stop);

}

}

void Display(const int arr[],const int size){

printf("Contents of array: ");

for(int i = 0 ; i < size ; i++)

printf("%-3d ",arr[i]);

printf("\n\n\n");

}

**OUTPUT:**

gml36:Assignment-1 cseb34$ ./Ex1

Enter the number of elements in the array: 10

Enter all the elements: 1 3 6 23 34 98 12 34 87 32

1 - Selection Sort

2 - Bubble Sort

3 - Linear Search

4 - Binary Search

Enter your choice: 1

Contents of array: 1 3 6 12 23 32 34 34 87 98

Do you want to stop 1/0: 0

Enter the number of elements in the array: 5

Enter all the elements: 1 23 34 92 34

1 - Selection Sort

2 - Bubble Sort

3 - Linear Search

4 - Binary Search

Enter your choice: 2

Contents of array: 1 23 34 34 92

Do you want to stop 1/0: 0

Enter the number of elements in the array: 7

Enter all the elements: 23 53 23 123 53 34 121

1 - Selection Sort

2 - Bubble Sort

3 - Linear Search

4 - Binary Search

Enter your choice: 3

Enter the number to search for: 123

Element found at index: 3

Do you want to stop 1/0: 0

Enter the number of elements in the array: 9

Enter all the elements: 1 312 23 43 12 52 89 23 41

1 - Selection Sort

2 - Bubble Sort

3 - Linear Search

4 - Binary Search

Enter your choice: 4

Enter the number to search for: 41

Contents of array: 1 12 23 23 41 43 52 89 312

Element found at index: 4

Do you want to stop 1/0: 1

//END OF PROGRAM