CSE-2202 : Algorithm

Lab Assignment– 01

Name : Enamul Haque

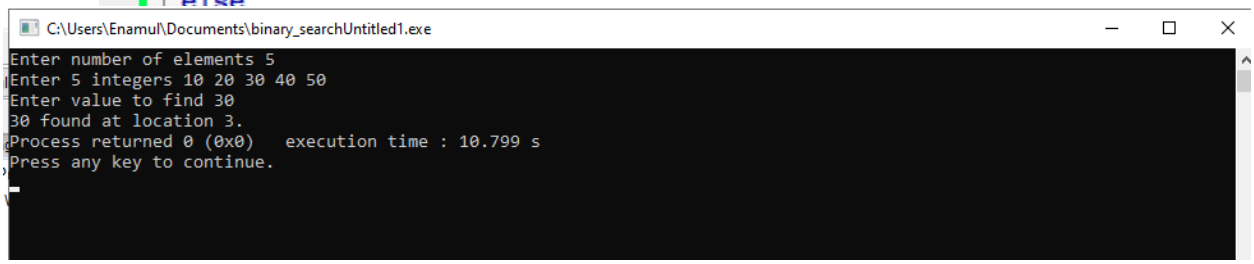
ID : 201631046089

Batch 46th

Bangladesh University

Problem 1: Binary Search

```c
#include <stdio.h>
int main()
{
int i, low, high, mid, n, key, array[100];
printf("Enter number of elements");
scanf("%d",&n);
printf("Enter %d integers", n);
for(i = 0; i < n; i++)
scanf("%d",&array[i]);
printf("Enter value to find");
scanf("%d", &key);
low = 0;
high = n - 1;
mid = (low+high)/2;
while (low <= high) {
if(array[mid] < key)
low = mid + 1;
else if (array[mid] == key) {
printf("%d found at location %d.", key, mid+1);
break;
}
else
```



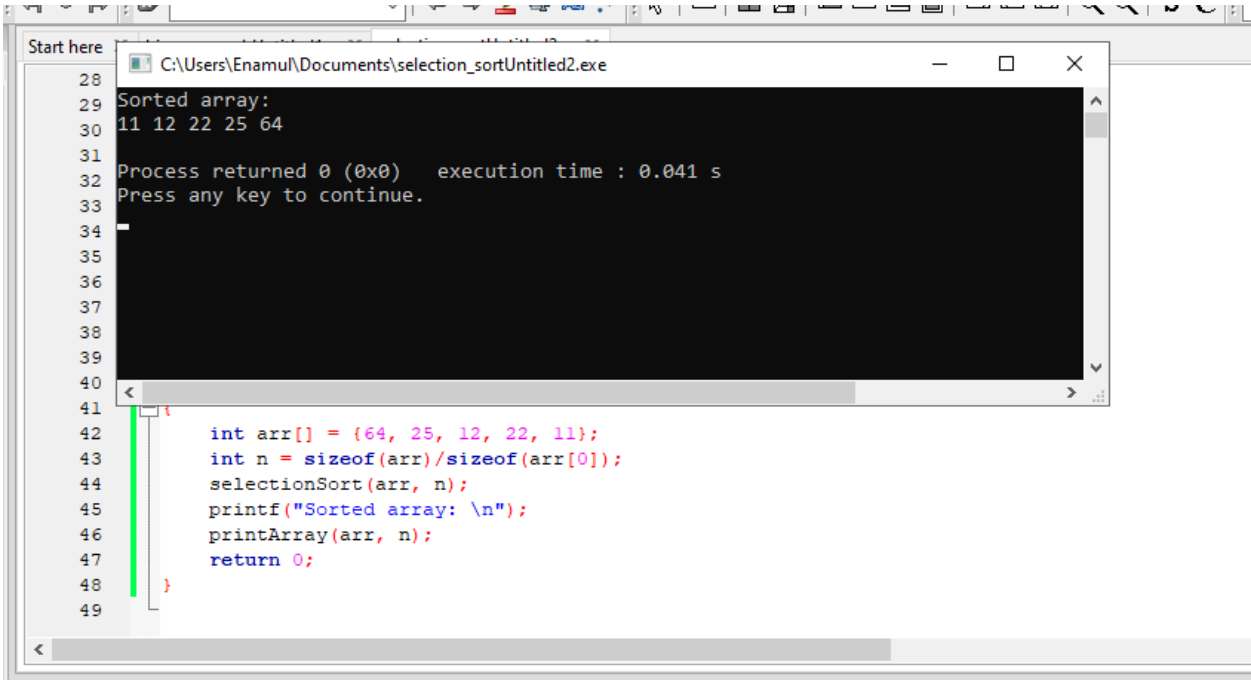
Problem 2: Selection sort

Start here | binary_searchUntitled1.c | selection_sortUntitled2.c

```c
// C program for implementation of selection sort
#include <stdio.h>

void swap(int *xp, int *yp)
{
    int temp = *xp;
    *xp = *yp;
    *yp = temp;
}

void selectionSort(int arr[], int n)
{
    int i, j, min_idx;

    // One by one move boundary of unsorted subarray
    for (i = 0; i < n-1; i++)
    {
        // Find the minimum element in unsorted array
        min_idx = i;
        for (j = i+1; j < n; j++)
        if (arr[j] < arr[min_idx])
            min_idx = j;
```

Start here | binary_searchUntitled1.c | selection_sortUntitled2.c

```c
        if(min_idx != i)
            swap(&arr[min_idx], &arr[i]);
    }
}

/* Function to print an array */
void printArray(int arr[], int size)
{
    int i;
    for (i=0; i < size; i++)
        printf("%d ", arr[i]);
    printf("\n");
}

// Driver program to test above functions
int main()
{
    int arr[] = {64, 25, 12, 22, 11};
    int n = sizeof(arr)/sizeof(arr[0]);
    selectionSort(arr, n);
    printf("Sorted array: \n");
    printArray(arr, n);
    return 0;
```


Start here
C:\Users\Enamul\Documents\selection_sortUntitled2.exe
Sorted array:
11 12 22 25 64
Process returned 0 (0x0) execution time : 0.041 s
Press any key to continue.
{
int arr[] = {64, 25, 12, 22, 11};
int n = sizeof(arr)/sizeof(arr[0]);
selectionSort(arr, n);
printf("Sorted array: \n");
printArray(arr, n);
return 0;
}