Output :- Binary Search

```
(base) aryankushwaha@Aryan-Kushwaha Lab-2 % cd "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/" &
 BinarySearch.c -o BinarySearch && "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/"BinarySearch
Enter Size of array: 3
Enter Array Elements: A[0] 10
A[1] 20
A[2] 30
Which number do you want to search: 30
Element is present at index 2
(base) aryankushwaha@Aryan-Kushwaha Lab-2 %
```

Output :- Heap Sort

```
 HeapSort.cpp -o HeapSort && "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/"HeapSort
HeapSort.cpp:48:16: warning: range-based for loop is a C++11 extension [-Wc++11-extensions]
    for (int i : arr)
               ^
1 warning generated.
Enter Size of array: 6
Enter Array Elements: A[0] 12
A[1] 32
A[2] 43
A[3] 7
A[4] 8
A[5] 9
Sorted array is
7 8 9 12 32 43
(base) aryankushwaha@Aryan-Kushwaha Lab-2 %
```

Output :- Merge Sort

```
(base) aryankushwaha@Aryan-Kushwaha Lab-2 % cd "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab
 MergeSort.c -o MergeSort && "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/"MergeSort
Enter Size of array: 6
Enter Array Elements: A[0]: 7
A[1]: 5
A[2]: 1
A[3]: 3
A[4]: 9
A[5]: 4
Given array is
7 5 1 3 9 4

Sorted array is
1 3 4 5 7 9
(base) aryankushwaha@Aryan-Kushwaha Lab-2 %
```

Output :- Quick Sort

```
(base) aryankushwaha@Aryan-Kushwaha Lab-2 % cd "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2
 QuickSort.cpp -o QuickSort && "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/"QuickSort
Enter Size of array: 6
Enter Array Elements: A[0]: 5
A[1]: 4
A[2]: 6
A[3]: 1
A[4]: 33
A[5]: 7
Given array:
5 4 6 1 33 7
Sorted array:
1 4 5 6 7 33
(base) aryankushwaha@Aryan-Kushwaha Lab-2 %
```

Output :- Selection in Expected Linear Time

```
(base) aryankushwaha@Aryan-Kushwaha Lab-2 % cd "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/" &&
 SelectionExp.cpp -o SelectionExp && "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/"SelectionExp
Enter Size of array: 6
Enter Array Elements: A[0]: 43
A[1]: 56
A[2]: 32
A[3]: 5
A[4]: 6
A[5]: 8
Enter k (1-based index of the smallest element to find): 3
K'th smallest element is 8
(base) aryankushwaha@Aryan-Kushwaha Lab-2 %
```

Output :- Selection in Worst Case Linear Time

```
sers/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/"SelectionWorst
(base) aryankushwaha@Aryan-Kushwaha Lab-2 % cd "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/" && g++
 SelectionWorst.cpp -o SelectionWorst && "/Users/Don't Open/5th Sem/Lab/Aaryan_28900/DAA/Lab-2/"SelectionWors
t
Enter Size of array: 5
Enter Array Elements: A[0]: 34
A[1]: 54
A[2]: 32
A[3]: 65
A[4]: 4
Enter k (1-based index of the smallest element to find): 2
K'th smallest element is 32
(base) aryankushwaha@Aryan-Kushwaha Lab-2 %
```