

PRACTICAL - 5

AIM : To perform

- (a) selection sort
- (b) bubble sort
- (c) insertion sort

(a) SELECTION SORT

PSEUDO CODE :

SelectionSort (Array A, n)

Step 1: for i from 0 to n-1, do steps 2-4
Step 2: Set min_index to i
Step 3: for j from i+1 to n, do step 4
Step 4: if A[j] < A[min_index], then set min_index to j
Step 5: Swap A[i] and A[min_index]
Step 6: End Loop

CODE :

```
#include <stdio.h>
void selectionSort(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {
        int minIndex = i;
        for (int j = i + 1; j < n; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;
            }
        }
        if (minIndex != i) {
            int temp = arr[i];
            arr[i] = arr[minIndex];
            arr[minIndex] = temp;
        }
    }
}
```

```
int main() {
    printf("Aabhas Kumar Jha - A2305221279\n\n");
    int n;
    printf("Enter the number of elements in the array: ");
    scanf("%d", &n);
    int arr[n];
    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    printf("Original array: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    selectionSort(arr, n);
    printf("\nSorted array: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

OUTPUT :

```
gcc 1000.c -o 1000
Aabhas Kumar Jha - A2305221279

Enter the number of elements in the array: 5
Enter the elements of the array:
12
0
132
-1
22
Original array: 12 0 132 -1 22
Sorted array: -1 0 12 22 132 %
```

COMPLEXITY : $O(n^2)$

(b) BUBBLE SORT

PSEUDO CODE :

BubbleSort (Array A, n)

Step 1: for i from 0 to n-1, do steps 2-4
Step 2: for j from 0 to n-i-1, do steps 3-4
Step 3: if A[j] > A[j+1], then swap A[j] and A[j+1]
Step 4: End Loop
Step 5: End Loop

CODE :

```
// bubble sort
```

```
#include <stdio.h>
```

```
void bubbleSort(int arr[], int n) {  
    for (int i = 0; i < n - 1; i++) {  
        for (int j = 0; j < n - i - 1; j++) {  
            if (arr[j] > arr[j + 1]) {  
                int temp = arr[j];  
                arr[j] = arr[j + 1];  
                arr[j + 1] = temp;  
            }  
        }  
    }  
}
```

```
int main() {  
    printf("Aabhas Kumar Jha - A2305221279\\n\\n");  
    int n;  
    printf("Enter the number of elements in the array: ");  
    scanf("%d", &n);
```

```

int arr[n];
printf("Enter the elements of the array:\n");
for (int i = 0; i < n; i++) {
    scanf("%d", &arr[i]);
}

printf("Original array: ");
for (int i = 0; i < n; i++) {
    printf("%d ", arr[i]);
}

bubbleSort(arr, n);

printf("\nSorted array: ");
for (int i = 0; i < n; i++) {
    printf("%d ", arr[i]);
}

return 0;
}

```

OUTPUT :

```

Aabhas Kumar Jha - A2305221279

Enter the number of elements in the array: 6
Enter the elements of the array:
-1
22
41
0
-2
43
Original array: -1 22 41 0 -2 43
Sorted array: -2 -1 0 22 41 43 %

```

COMPLEXITY : $O(n^2)$

(c) INSERTION SORT

PSEUDO CODE :

InsertionSort (Array A, n)

Step 1: for i from 1 to n-1, do steps 2-5
Step 2: Set key to A[i]
Step 3: Set j to i-1
Step 4: while j >= 0 and A[j] > key, do step 5
Step 5: Set A[j+1] to A[j] and decrement j
Step 6: Set A[j+1] to key
Step 7: End Loop

}

CODE :

```
// insertion sort
```

```
#include <stdio.h>
```

```
void insertionSort(int arr[], int n) {  
    for (int i = 1; i < n; i++) {  
        int key = arr[i];  
        int j = i - 1;  
        while (j >= 0 && arr[j] > key) {  
            arr[j + 1] = arr[j];  
            j--;  
        }  
        arr[j + 1] = key;  
    }  
}  
  
int main() {  
    printf("Aabhas Kumar Jha - A2305221279\\n\\n");  
    int n;  
    printf("Enter the number of elements in the array: ");
```

Y

```
scanf("%d", &n);

int arr[n];
printf("Enter the elements of the array:\n");
for (int i = 0; i < n; i++) {
    scanf("%d", &arr[i]);
}
```

```
printf("Original array: ");
for (int i = 0; i < n; i++) {
    printf("%d ", arr[i]);
}
```

```
insertionSort(arr, n);
```

```
printf("\nSorted array: ");
for (int i = 0; i < n; i++) {
    printf("%d ", arr[i]);
}

return 0;
}
```

OUTPUT :

```
Aabhas Kumar Jha - A2305221279

Enter the number of elements in the array: 4
Enter the elements of the array:
22
0
12
-1
Original array: 22 0 12 -1
Sorted array: -1 0 12 22 %
```

COMPLEXITY : $O(n^2)$