### **Selection Sort Program:**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void selectionSort(int arr[], int n) {

int i, j, min\_idx;

for (i = 0; i < n - 1; i++) {

min\_idx = i;

for (j = i + 1; j < n; j++) {

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

min\_idx = j;

}

int temp = arr[min\_idx];

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

arr[i] = temp;

}

}

int main() {

int n;

printf("Enter the size of the array: ");

scanf("%d", &n);

int arr[n];

// Generate 'n' random numbers

srand(time(NULL));

printf("Randomly generated array:\n");

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

arr[i] = rand() % 100; // Generate random numbers between 0 and 99

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

}

clock\_t start\_time = clock();

selectionSort(arr, n);

clock\_t end\_time = clock();

double selectionSortTime = ((double) (end\_time - start\_time)) / CLOCKS\_PER\_SEC;

printf("\nSelection Sort:\n");

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

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

printf("\nTime taken: %f seconds\n", selectionSortTime);

return 0;

}

### **Insertion Sort Program:**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void insertionSort(int arr[], int n) {

int i, key, j;

for (i = 1; i < n; i++) {

key = arr[i];

j = i - 1;

while (j >= 0 && arr[j] > key) {

arr[j + 1] = arr[j];

j = j - 1;

}

arr[j + 1] = key;

}

}

int main() {

int n;

printf("Enter the size of the array: ");

scanf("%d", &n);

int arr[n];

// Generate 'n' random numbers

srand(time(NULL));

printf("Randomly generated array:\n");

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

arr[i] = rand() % 100; // Generate random numbers between 0 and 99

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

}

clock\_t start\_time = clock();

insertionSort(arr, n);

clock\_t end\_time = clock();

double insertionSortTime = ((double) (end\_time - start\_time)) / CLOCKS\_PER\_SEC;

printf("\nInsertion Sort:\n");

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

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

printf("\nTime taken: %f seconds\n", insertionSortTime);

return 0;

}