S.No: 11	Exp. Name: ***Selection Sort***	Date:2025-10-18

Aim:

Write a C program that prompts the user to enter the size of an array and its elements. The program should then sort the array in ascending order using the Selection Sort algorithm and display the sorted array.

Input Format:

- The first line of input contains an integer $size$ representing the size of the array.
- The second line of input contains $size$ integers separated by spaces, representing the elements of the array.

Output Format:

- The first line prints the original array as:

```
Original array: <element1> <element2> ... <elementN>
```

- The second line prints the sorted array as:

```
Sorted array: <element1> <element2> ... <elementN>
```

Source Code:

selection.c

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

void selectionSort(int arr[], int size) {
    for ( int i = 0; i<size-1; i++ ) {
        int minIndex = i;

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

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

                minIndex = j;
            }
        }

        int temp = arr[i];

        arr[i] = arr[minIndex];

        arr[minIndex] = temp;

    }
}

void displayArray(int arr[], int size) {
    for(int i = 0; i<size; i++)
      printf("%d ", arr[i]);
   printf("\n");
}
```

```
int main() {
    int size;

    printf("Enter the size of the array: ");
    scanf("%d", &size);

    int *arr = (int *)malloc(size * sizeof(int));

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

    printf("Original array: ");
    displayArray(arr, size);

    selectionSort(arr, size);
    printf("Sorted array: ");
    displayArray(arr, size);
    free(arr);

    return 0;
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter the size of the array: 4
Enter 4 elements of the array: 2 1 6 9
Original array: 2 1 6 9
Sorted array: 1 2 6 9

Test Case - 2
User Output
Enter the size of the array: 5
Enter 5 elements of the array: 67 3 0 23 7
Original array: 67 3 0 23 7
Sorted array: 0 3 7 23 67