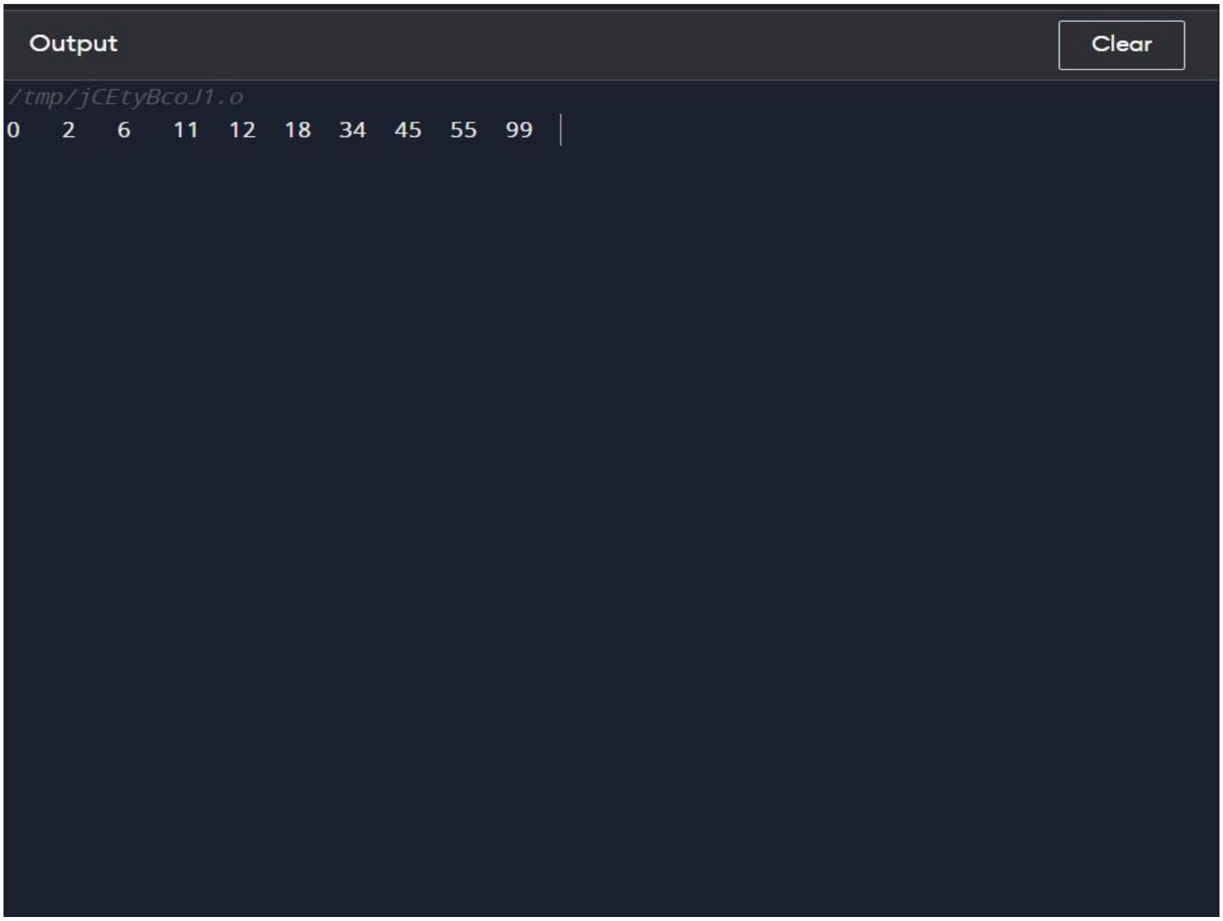
Practical No:1 Introduction

Selection Sort

```
#include <stdio.h>
int main() {
  int arr[10]={6,12,0,18,11,99,55,45,34,2};
  int n=10;
  int i, j, position, swap;
  for (i = 0; i < (n - 1); i++) {
    position = i;
    for (j = i + 1; j < n; j++) {
      if (arr[position] > arr[j])
        position = j;
    }
    if (position != i) {
      swap = arr[i];
      arr[i] = arr[position];
      arr[position] = swap;
    }
  }
  for (i = 0; i < n; i++)
    printf("%d\t", arr[i]);
  return 0;
}
```

OutPut:

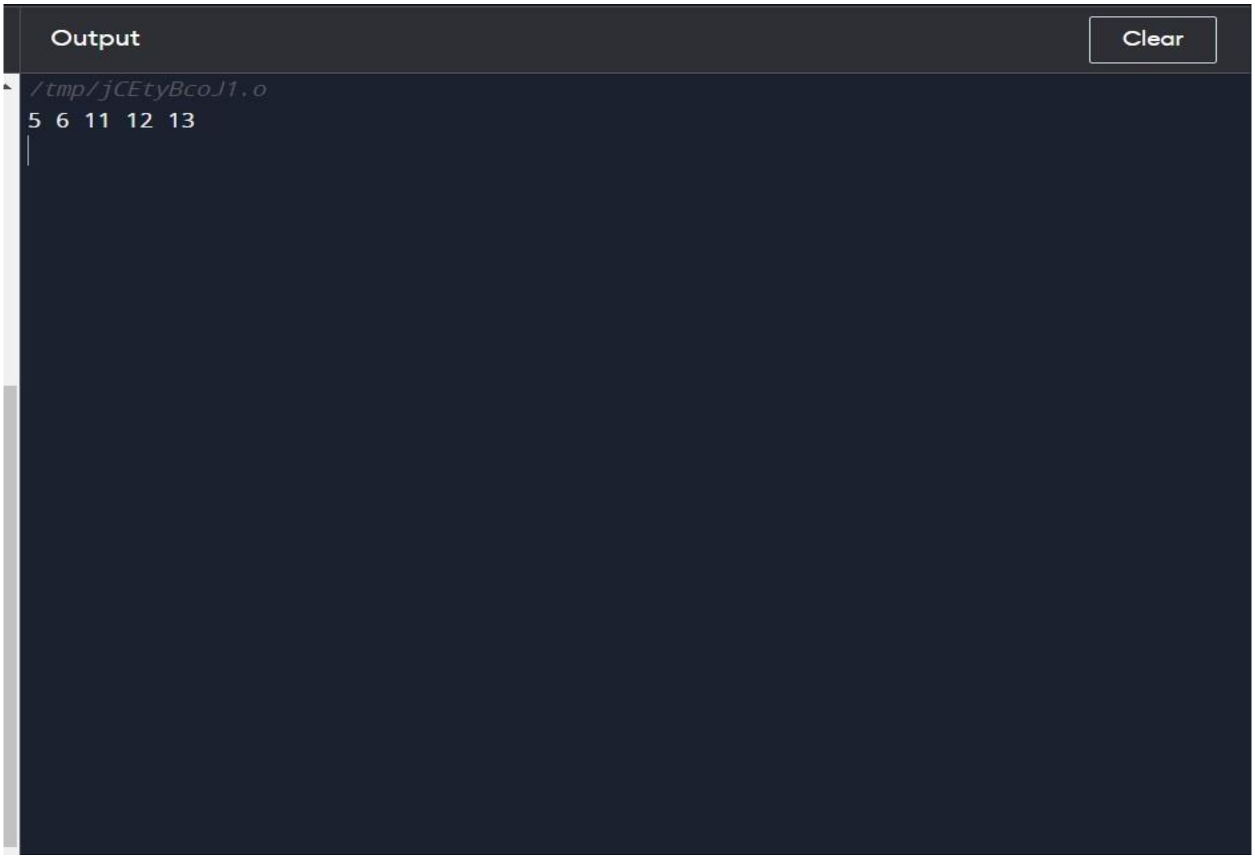


```
/tmp/jCEtyBcoJ1.o
0   2   6   11  12  18  34  45  55  99
```

Insertion Sort

```
#include <math.h>

#include <stdio.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;

    }

}

void printArray(int arr[], int n)

{

    int i;

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

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

    printf("\n");

}

int main()

{

    int arr[] = { 12, 11, 13, 5, 6 };

    int n = sizeof(arr) / sizeof(arr[0]);

    insertionSort(arr, n);

    printArray(arr, n);

    return 0;}
```

OutPut:



```
/tmp/jCEtyBcoJ1.o
5 6 11 12 13
```