**NON-RECURSIVE INSERTION SORT**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <time.h>

void Insert\_Sort(int [], int);

int main()

{

clock\_t start,end;

double time;

int arr[100000], n, i;

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

scanf("%d", &n);

printf("Enter the elements in the array:\n");

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

{

arr[i] = rand()%100;

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

}

start = clock();

Insert\_Sort(arr, n);

end = clock();

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

printf("\nTime taken : %lf\n",time);

printf("\nThe sorted list in ascending order is\n");

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

{

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

}

return 0;

}

void Insert\_Sort(int arr[], int n)

{

int i, search, j;

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

search = arr[i];

j = i - 1;

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

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

j = j - 1;

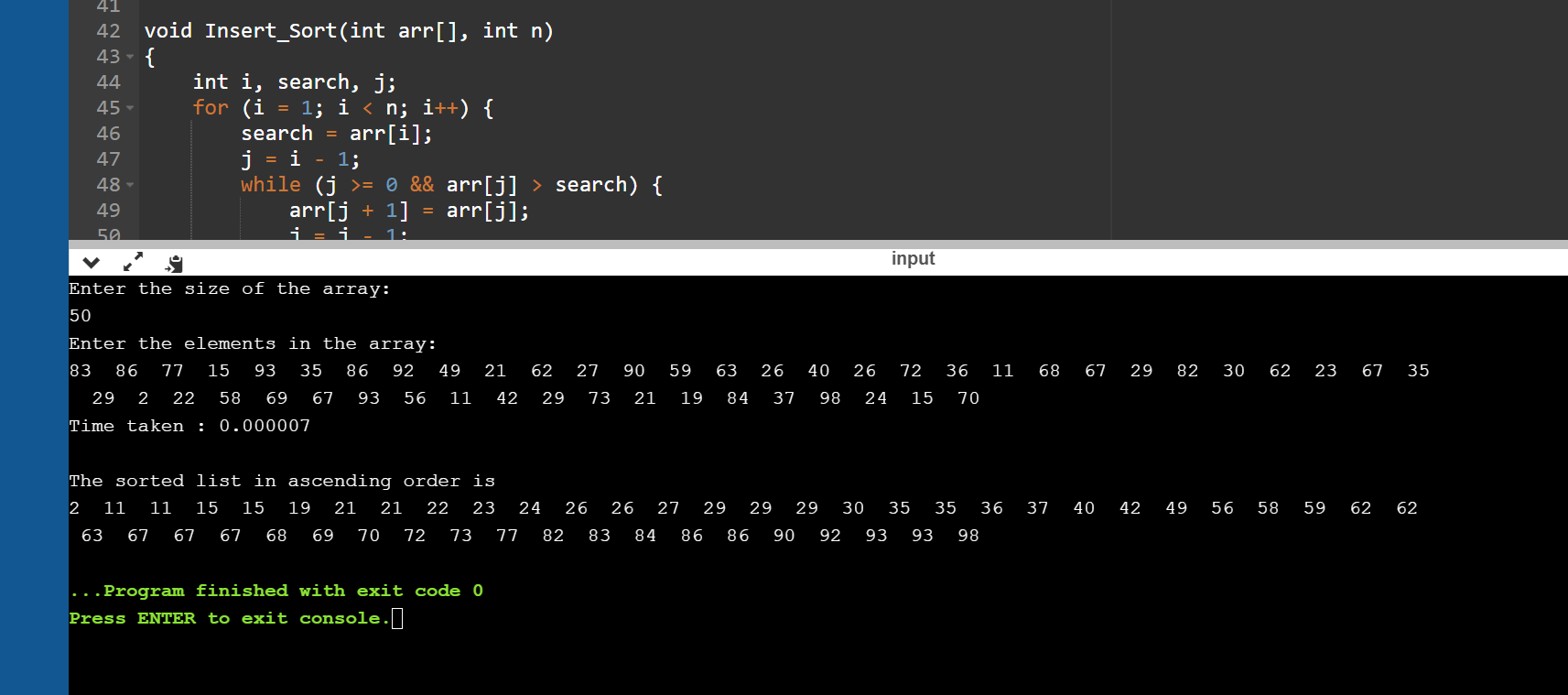
}

arr[j + 1] = search;

}

}

OUTPUT:



|  |  |
| --- | --- |
| N | Time Taken |
| 50 | 0.000007 |
| 100 | 0.000015 |
| 200 | 0.000065 |
| 500 | 0.000312 |
| 1000 | 0.001089 |
| 5000 | 0.026088 |
| 10000 | 0.087888 |
| 15000 | 0.183433 |
| 20000 | 0.40368 |
| 25000 | 0.537591 |

**RECURSIVE INSERTION SORT**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void Insertion\_Sort(int [], int);

int main()

{

clock\_t start,end;

double time;

int arr[100000], n, i;

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

scanf("%d", &n);

printf("Enter the elements in the array:\n");

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

{

arr[i] = rand()%100;

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

}

start = clock();

Insertion\_Sort(arr, n);

end = clock();

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

printf("\nTime taken : %lf\n",time);

printf("\nThe sorted list in ascending order is\n");

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

{

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

}

return 0;

}

void Insertion\_Sort(int arr[], int n){

if (n <= 1)

return;

Insertion\_Sort( arr, n-1 );

int last = arr[n-1];

int j = n-2;

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

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

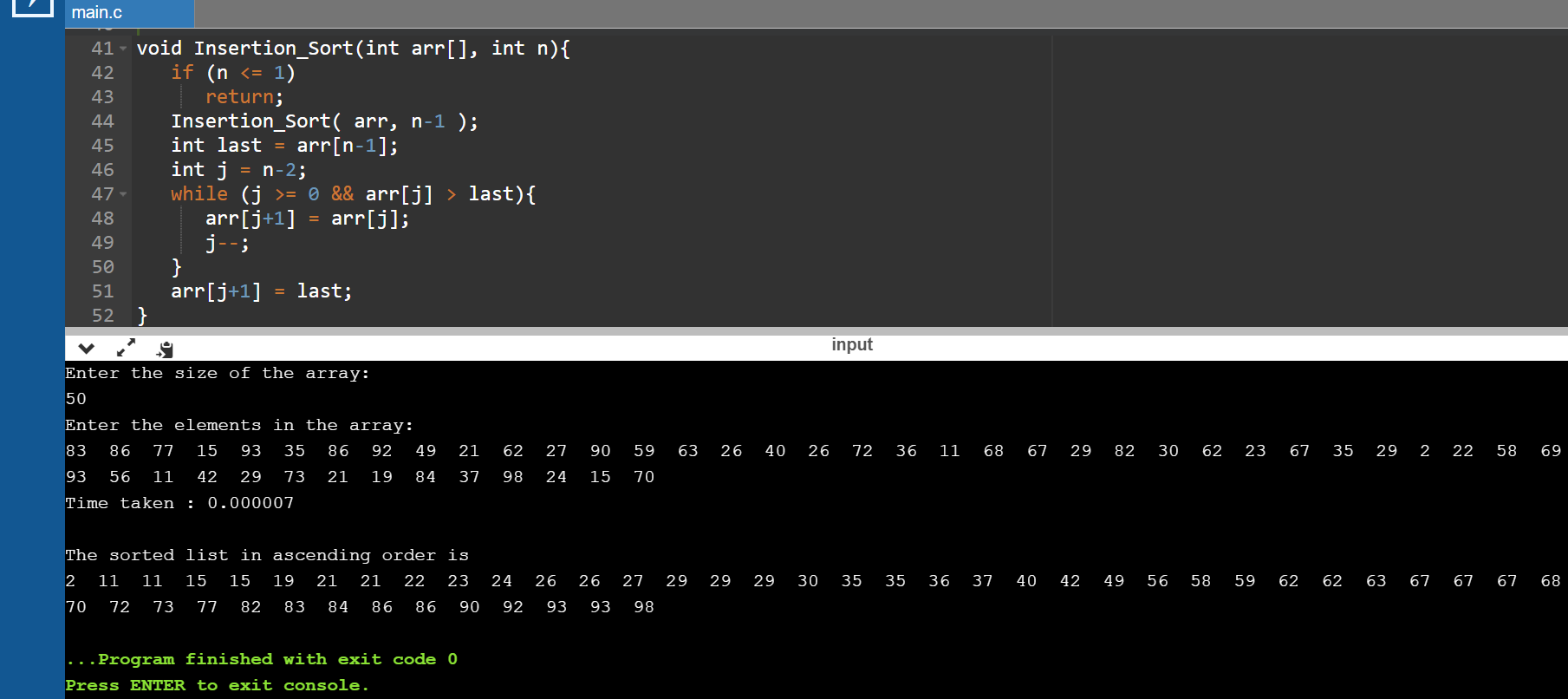
j--;

}

arr[j+1] = last;

}

OUTPUT:



|  |  |
| --- | --- |
| N | Time Taken |
| 50 | 0.000007 |
| 100 | 0.000024 |
| 200 | 0.000063 |
| 500 | 0.000341 |
| 1000 | 0.001115 |
| 5000 | 0.024857 |
| 10000 | 0.088347 |
| 15000 | 0.189293 |
| 20000 | 0.403715 |
| 25000 | 0.580936 |