

[Geeks Classes](#)[Login](#)[Write an Article](#)

Recursive Insertion Sort

Insertion sort is a simple sorting algorithm that works the way we sort playing cards in our hands.

Below is an iterative algorithm for insertion sort

Algorithm

```
// Sort an arr[] of size n
insertionSort(arr, n)
    Loop from i = 1 to n-1.
        a) Pick element arr[i] and insert
           it into sorted sequence arr[0..i-1]
```

Example:



Refer [Insertion Sort](#) for more details.

How to implement it recursively?

Recursive Insertion Sort has no performance/implementation advantages, but can be a good question to check one's understanding of Insertion Sort and recursion.

If we take a closer look at Insertion Sort algorithm, we keep processed elements sorted and insert new elements one by one in the inserted array.

Recursion Idea.

1. Base Case: If array size is 1 or smaller, return.
2. Recursively sort first n-1 elements.
3. Insert last element at its correct position in sorted array.

Below is implementation of above idea.

C/C++

```
// Recursive C++ program for insertion sort
#include <iostream>
using namespace std;

// Recursive function to sort an array using
// insertion sort
void insertionSortRecursive(int arr[], int n)
{
    // Base case
    if (n <= 1)
        return;

    // Sort first n-1 elements
    insertionSortRecursive( arr, n-1 );

    // Insert last element at its correct position
    // in sorted array.
    int last = arr[n-1];
    int j = n-2;

    /* Move elements of arr[0..i-1], that are
    greater than key, to one position ahead
    of their current position */
    while (j >= 0 && arr[j] > last)
    {
        arr[j+1] = arr[j];
        j--;
    }
    arr[j+1] = last;
}
```

```
// A utility function to print an array of size n
void printArray(int arr[], int n)
{
    for (int i=0; i < n; i++)
        cout << arr[i] <<" ";
}

/* Driver program to test insertion sort */
int main()
{
    int arr[] = {12, 11, 13, 5, 6};
    int n = sizeof(arr)/sizeof(arr[0]);

    insertionSortRecursive(arr, n);
    printArray(arr, n);

    return 0;
}
```

Java

```
// Recursive Java program for insertion sort

import java.util.Arrays;

public class GFG
{
    // Recursive function to sort an array using
    // insertion sort
    static void insertionSortRecursive(int arr[], int n)
    {
        // Base case
        if (n <= 1)
            return;

        // Sort first n-1 elements
        insertionSortRecursive( arr, n-1 );

        // Insert last element at its correct position
        // in sorted array.
        int last = arr[n-1];
        int j = n-2;

        /* Move elements of arr[0..i-1], that are
        greater than key, to one position ahead
        of their current position */
        while (j >= 0 && arr[j] > last)
        {
            arr[j+1] = arr[j];
            j--;
        }
        arr[j+1] = last;
    }
}
```

```
// Driver Method
public static void main(String[] args)
{
    int arr[] = {12, 11, 13, 5, 6};

    insertionSortRecursive(arr, arr.length);

    System.out.println(Arrays.toString(arr));
}
}
```

Python

```
# Recursive Python program for insertion sort
# Recursive function to sort an array using insertion sort

def insertionSortRecursive(arr,n):
    # base case
    if n<=1:
        return

    # Sort first n-1 elements
    insertionSortRecursive(arr,n-1)
    '''Insert last element at its correct position
    in sorted array.'''
    last = arr[n-1]
    j = n-2

    # Move elements of arr[0..i-1], that are
    # greater than key, to one position ahead
    # of their current position
    while (j>=0 and arr[j]>last):
        arr[j+1] = arr[j]
        j = j-1

    arr[j+1]=last

# A utility function to print an array of size n
def printArray(arr,n):
    for i in range(n):
        print arr[i],

# Driver program to test insertion sort
arr = [12,11,13,5,6]
n = len(arr)
insertionSortRecursive(arr, n)
printArray(arr, n)

# Contributed by Harsh Valecha
```

Output :

5 6 11 12 13

Recursive Insertion Sort | GeeksforGeeks



This article is contributed by **Sahil Chhabra (akku)**. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Practice Tags : [Recursion](#) [Sorting](#)

Article Tags : [Recursion](#) [Sorting](#)

[Login to Improve this Article](#)

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Recommended Posts:

[Merge Sort](#)

[Insertion Sort](#)

[Recursive Bubble Sort](#)

Selection Sort**Iterative Merge Sort**

Find n-th node in Postorder traversal of a Binary Tree

Convert a Binary Tree such that every node stores the sum of all nodes in its right subtree

Recursive program to generate power set

Check for balanced parenthesis without using stack

Smallest number in BST which is greater than or equal to N

(Login to Rate)

2.2Average Difficulty : **2.2/5.0**
Based on **20** vote(s)☐

Add to TODO List

☐

Mark as DONE

Basic

Easy

Medium

Hard

Expert

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments

Share this post!

A computer science portal for geeks

710-B, Advant Navis Business Park,
Sector-142, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org**COMPANY**About Us
Careers
Privacy Policy
Contact Us**PRACTICE**Company-wise
Topic-wise
Contests
Subjective Questions**LEARN**Algorithms
Data Structures
Languages
CS Subjects
Video Tutorials**CONTRIBUTE**Write an Article
Write Interview Experience
Internships
Videos

@geeksforgeeks, Some rights reserved