A computer science portal for geeks

Amazon Internship Interview Experience

Sort an array without changing position of negative numbers

Find the longest string that can be made up of other strings from the array

Count the triplets such that A[i] < B[j] < C[k]

Comparison among Bubble Sort, Selection Sort and Insertion Sort

Two nodes of a BST are swapped, correct the BST | Set-2

Nagarro Interview Experience Off-campus

Greatest contiguous sub-array of size K

Bubble Sort for Linked List by Swapping nodes

Merge two BSTs with constant extra space

Maximum water that can be stored between two buildings

Pair with largest sum which is less than K in the array

Find a triplet in an array whose sum is closest to a given number

Keep track of previous indexes after sorting a vector in C++ STI

Find maximum meetings in one

Range Queries to Find number of sub-arrays with a given xor

Split the array elements into strictly increasing and decreasing sequence

Count pairs with given sum | Set 2

Iterative selection sort for linked list

Maximum Length Chain of Pairs I Set-2

Find the number of elements greater than k in a sorted array

Program to print an array in Pendulum Arrangement with constant space

Sort an array of strings based on the frequency of good words in them

TimSort

TimSort is a sorting algorithm based on Insertion Sort and Merge Sort.

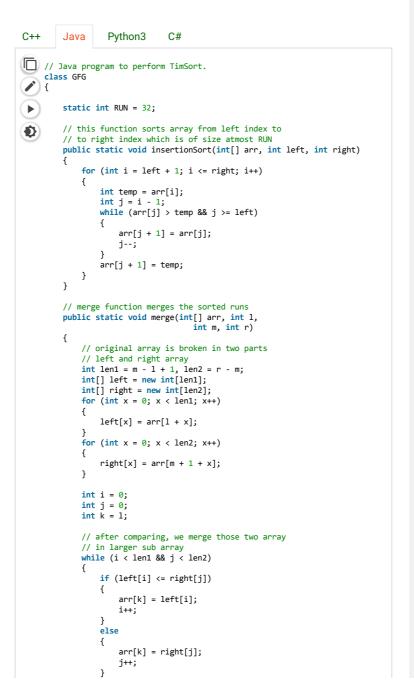
- 1. A stable sorting algorithm works in O(n Log n) time
- 2. Used in Java's Arrays.sort() as well as Python's sorted() and sort().
- First sort small pieces using Insertion Sort, then merges the pieces using merge of merge sort.

We divide the Array into blocks known as **Run**. We sort those runs using insertion sort one by one and then merge those runs using combine function used in merge sort. If the size of Array is less than run, then Array get sorted just by using Insertion Sort. The size of run may vary from 32 to 64 depending upon the size of the array. Note that merge function performs well when sizes subarrays are powers of 2. The idea is based on the fact that insertion sort performs well for small arrays.

Recommended: Please try your approach on <u>{IDE}</u> first, before moving on to the solution.

Details of below implementation :

- We consider size of run as 32.
- We one by one sort pieces of size equal to run
- After sorting individual pieces, we merge them one by one. We double the size of merged subarrays after every iteration.





Most popular in Sorting

Find the winner of the match | Multiple Queries

Product of minimum edge weight between all pairs of a Tree

Merge K sorted Doubly Linked List in Sorted Order

Check if the string contains consecutive letters and each letter occurs exactly once

Maximal Disjoint Intervals

More related articles in Sorting

Rearrange the characters of the string such that no two adjacent characters are

Find minimum changes required in an array for it to contain k distinct elements

IntroSort or Introspective sort

Remove elements to make array sorted

Unbounded Fractional Knapsack

Divide array into two parts with equal sum according to the given constraints

Sort an array of strings according to string lengths using Map

Check whether an array can be made strictly decreasing by modifying at most one element



```
k++;
         }
         \ensuremath{//} copy remaining elements of left, if any
         while (i < len1)
             arr[k] = left[i];
         }
         // copy remaining element of right, if any
         while (j < len2)
             arr[k] = right[j];
             j++;
         }
    }
     // iterative Timsort function to sort the
     // array[0...n-1] (similar to merge sort)
     public static void timSort(int[] arr, int n)
         // Sort individual subarrays of size RUN
for (int i = 0; i < n; i += RUN)</pre>
             insertionSort(arr, i, Math.min((i + 31), (n - 1)));
         }
         // start merging from size RUN (or 32). It will merge
         // to form size 64, then 128, 256 and so on ...
for (int size = RUN; size < n; size = 2 * size)
             // pick starting point of left sub array. We
             // are going to merge arr[left..left+size-1]
             // and arr[left+size, left+2*size-1]
             // After every merge, we increase left by 2*size for (int left = 0; left < n; left += 2 * size)
                  // find ending point of left sub array
                  // mid+1 is starting point of right sub array
                  int mid = left + size - 1;
                  int right = Math.min((left + 2 * size - 1), (n - 1));
                  // merge sub array arr[left....mid] &
                  // arr[mid+1....right]
                  merge(arr, left, mid, right);
        }
    }
     // utility function to print the Array
    public static void printArray(int[] arr, int n)
         for (int i = 0; i < n; i++)
         {
             System.out.print(arr[i] + " ");
         System.out.print("\n");
    }
    // Driver code
    public static void main(String[] args)
         int[] arr = {5, 21, 7, 23, 19};
         int n = arr.length;
         System.out.print("Given Array is\n");
         printArray(arr, n);
         timSort(arr, n);
         System.out.print("After Sorting Array is\n");
         printArray(arr, n);
}
// This code has been contributed by 29AjayKumar
```

Output:

```
Given Array is
5 21 7 23 19
After Sorting Array is
5 7 19 21 23
```

References:

https://svn.python.org/projects/python/trunk/Objects/listsort.txt https://en.wikipedia.org/wiki/Timsort#Minimum_size_.28minrun.29

This article is contributed by Aditya Kumar. 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

consecutive English alphabets

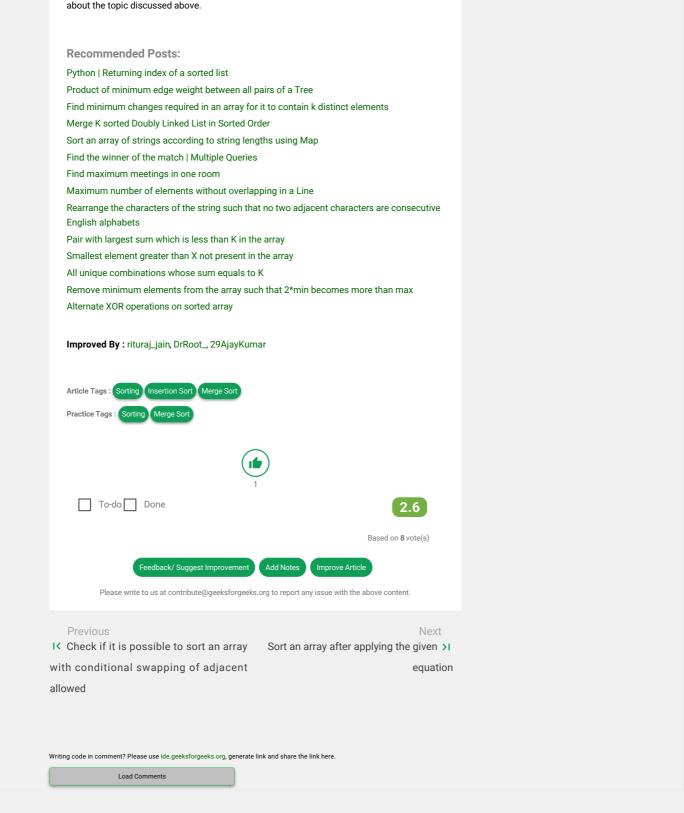
Find the kth smallest number with sum of digits as m

Alternate XOR operations on sorted array

Maximum number of elements without overlapping in a Line

Sort ugly numbers in an array at their relative positions







5th Floor, A-118, Sector-136, Noida, Uttar Pradesh - 201305 feedback@geeksforgeeks.org COMPANY

About Us Careers Privacy Policy Contact Us LEARN

Algorithms Data Structures Languages CS Subjects Video Tutorials PRACTICE

Courses Company-wise Topic-wise How to begin? CONTRIBUTE

Write an Article Write Interview Experience Internships Videos

e use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood of

Cookie Policy & Privacy Policy