



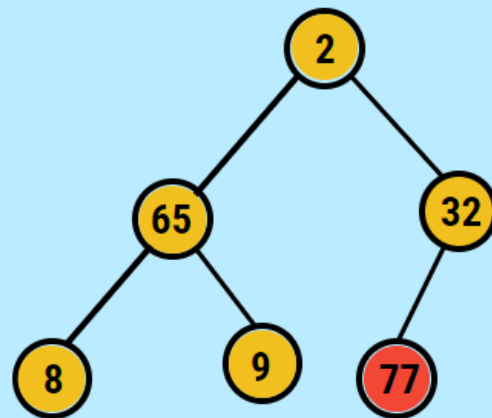
(<https://www.educba.com/software-development/>)



(<https://www.educba.com/sorting-in-c/>)

→ (<https://www.educba.com/bubble-sort-in-c/>)

# Heap Sort in C



Numbers

2	8	9	32	65	77
---	---	---	----	----	----

[www.educba.com](https://www.educba.com)

## Introduction to Heap Sort in C

Sorting is a technique that is all about the ordering of elements based on different properties (Properties like arranging data in ascending, descending or alphabetical orders). One more example of sorting that we can think of here is the ordering of items during online shopping.



We can relate to prices, popularity, latest and so on. So there are many techniques for this



[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

[programming language \(https://www.educba.com/software-development/c-programming-language\)](https://www.educba.com/software-development/c-programming-language)

### Start Your Free Software Development Course

Web development, programming languages, Software testing & others

## The logic for Heap Sort

How actually can we perform heap sort? Let's check out below.

Firstly, the heap is one of the tree-based data structure. The tree involved here is always a Complete Binary Tree. And, there are two kinds of heap

- **Min – Heap:** Generally arranged in ascending order, that is if the parent node element is having a value less than that of child node elements.
- **Max – Heap:** Generally arranged in descending order, that is if the parent node element is having value more than that of child node elements.

## Steps for Heap Sort

- Once an unsorted list data is obtained, elements are organized in the heap data structure either based on creating a min-heap or a max-heap.
- The first element from the above list is added into our array
- Again forming the heap data structure technique same as the first step is followed and again either the highest element or the least element is picked up and added into array.
- Repeated steps help us getting the array with the sorted list.



## Program for Heap Sort in C



[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
int h[20],num,i,j,root,t,x;
printf("Enter number of elements :");
scanf("%d", &num);
printf("\nEnter the elements : ");
for (i = 0; i < num; i++)
scanf("%d", &h[i]);
// build max heap
for(i=0;i<num;i++)
{
x=i;
do
{
root = (x - 1) / 2;
if (h[root] < h[x])
{
t = h[root];
h[root] = h[x];
h[x] = t;
}
x = root;
} while (x != 0);
}
printf("Heap array formed is: ");
for (i = 0; i < num; i++)
printf("%d\t ", h[i]);

for (j = num - 1; j >= 0; j--)
```





(<https://www.educba.com/software-development/>)

```
root = 0;
do
{
x = 2 * root + 1;
if ((h[x] < h[x + 1]) && x < j-1)
x++;
if (h[root]<h[x] && x<j)
{
t = h[root];
h[root] = h[x];
h[x] = t;
}
root = x;
} while (x < j);
}
printf("\nThe sorted array is : ");
for (i = 0; i < num; i++)
printf("\t %d", h[i]);
}
```

First, we are asking the user to input the number of elements that are taken for sorting and then the user is allowed to enter different elements that are to be sorted.

🔗 Popular Course in this category





(<https://www.educba.com/software-development/>)

★★★★★ 4.5 (8,612 ratings)

Course Price

**\$79** ~~\$399~~

**View Course**

(<https://www.educba.com/software-development/courses/c-programming-course/?btnz=edu-blg-inline-banner1>)

### Related Courses

C++ Training (4 Courses, 5 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/c-course/?btnz=edu-blg-inline-banner1>)

Java Training (40 Courses, 29 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/java-course/?btnz=edu-blg-inline-banner1>)

### Steps Followed

- The next which we are focusing on is to create a heap array, in this case, max-heap array.
- The main condition for getting a max – heap array is to check that no parent node value is less than its child node value. We are going to swap until we achieve that condition.
- The main advantage in this complete binary tree is, the left and right child nodes of a parent node can be accessed with values  $2(i) + 1$  and  $2*(i) + 2$  values respectively. Where  $i$  is the parent node.
- So, through that way here, we are placing our root node which contains the maximum value in the rightmost leaf node place. And then again following the same procedure such that the next maximum number now becomes the root node.
- We are going to follow the same procedure until only one node is left in the heap
- And then, we are arranging our heap array to form a perfect sorted array in increasing





(<https://www.educba.com/software-development/>)

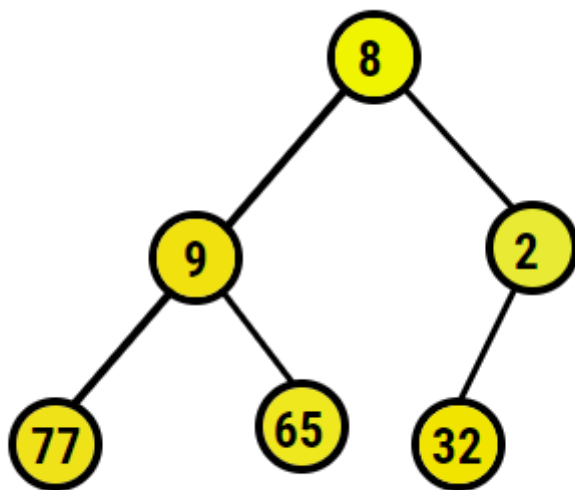
The output is attached below.

Let me show you the pictorial representation of the happenings:

- The data entered is first represented in the form of a single-dimensional array as follows.

Numbers	8	9	2	77	65	32
Array h	h[0]	h[1]	h[2]	h[3]	h[4]	h[5]

- The pictorial representation of the binary tree formed is as follows:



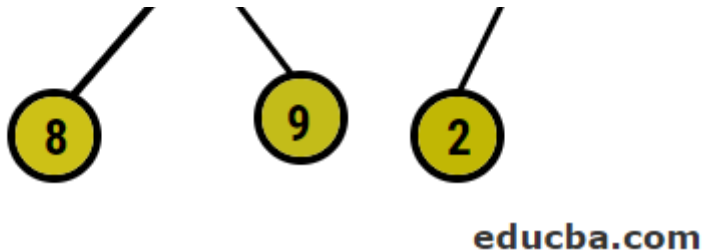
educba.com

- Now, we are going to convert to the max heap by making sure that all the parent nodes are always greater than child nodes. As mentioned in the output under heap sorted array, the pictorial representation would be:

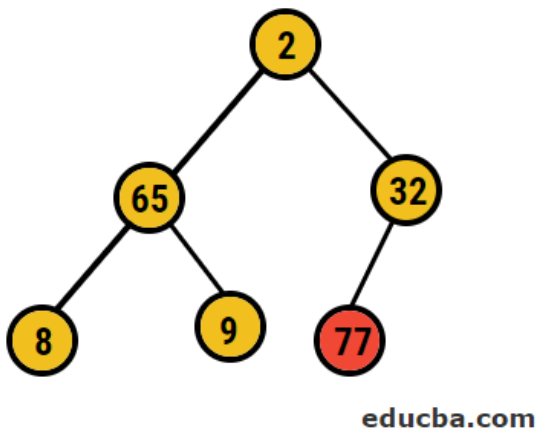




(<https://www.educba.com/software-development/>)



- After this, we are going to swap the root node with the extreme leaf node and then delete it from the tree. The leaf node would be the root now and again same process is followed to again get the highest element in the root



- So, in this case, 77 digits are being deleted from this tree and placed into our sorted array and the process is repeated.

The above we have seen it for forming max heap array. The same process is dealt with with the min-heap array formation also. As discussed above, the only difference is with the relationship between parent and child node elements.

As an exercise, can you try forming the heap sort in descending order?



## Conclusion



[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

## Recommended Articles

This is a guide to Heap Sort in C. Here we discuss the logic and the Steps for Heap Sort with the sample code and output along with pictorial representations. You may also have a look at the following articles to learn more –

1. [Heap Sort In Java \(https://www.educba.com/heap-sort-in-java/\)](https://www.educba.com/heap-sort-in-java/)
2. [Selection Sort in Java \(https://www.educba.com/selection-sort-in-java/\)](https://www.educba.com/selection-sort-in-java/)
3. [Heap Sort in C++ \(https://www.educba.com/heap-sort-in-c-plus-plus/\)](https://www.educba.com/heap-sort-in-c-plus-plus/)
4. [Heap Sort in Python \(https://www.educba.com/heap-sort-in-python/\)](https://www.educba.com/heap-sort-in-python/)

## C PROGRAMMING TRAINING (3 COURSES, 5 PROJECT)

- ☒ 3 Online Courses
- ☒ 5 Hands-on Projects
- ☒ 34+ Hours
- ☒ Verifiable Certificate of Completion
- ☒ Lifetime Access

**Learn More**

<https://www.educba.com/software-development/courses/c-programming-course/?btnz=educba-inline-banner3>







[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

## About Us

Blog (<https://www.educba.com/blog/?source=footer>)

Who is EDUCBA? (<https://www.educba.com/about-us/?source=footer>)

Sign Up (<https://www.educba.com/software-development/signup/?source=footer>)

Corporate Training (<https://www.educba.com/corporate/?source=footer>)

Certificate from Top Institutions (<https://www.educba.com/educbalive/?source=footer>)

Contact Us (<https://www.educba.com/contact-us/?source=footer>)

Verifiable Certificate (<https://www.educba.com/software-development/verifiable-certificate/?source=footer>)

Reviews (<https://www.educba.com/software-development/reviews/?source=footer>)

Terms and Conditions (<https://www.educba.com/terms-and-conditions/?source=footer>)

Privacy Policy (<https://www.educba.com/privacy-policy/?source=footer>)

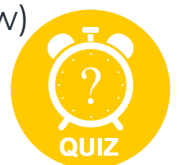
## Apps

iPhone & iPad (<https://itunes.apple.com/in/app/educba-learning-app/id1341654580?mt=8>)

Android (<https://play.google.com/store/apps/details?id=com.educba.www>)

## Resources

Free Courses (<https://www.educba.com/software-development/free-courses/?source=footer>)





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

development-tutorials/?source=footer)

## Certification Courses

All Courses (<https://www.educba.com/software-development/courses/?source=footer>)

Software Development Course - All in One Bundle  
(<https://www.educba.com/software-development/courses/software-development-course/?source=footer>)

Become a Python Developer (<https://www.educba.com/software-development/courses/python-certification-course/?source=footer>)

Java Course (<https://www.educba.com/software-development/courses/java-course/?source=footer>)

Become a Selenium Automation Tester (<https://www.educba.com/software-development/courses/selenium-training-certification/?source=footer>)

Become an IoT Developer (<https://www.educba.com/software-development/courses/iot-course/?source=footer>)

ASP.NET Course (<https://www.educba.com/software-development/courses/asp-net-course/?source=footer>)

VB.NET Course (<https://www.educba.com/software-development/courses/vb-net-course/?source=footer>)

PHP Course (<https://www.educba.com/software-development/courses/php-course/?source=footer>)

© 2022 - EDUCBA. ALL RIGHTS RESERVED. THE CERTIFICATION NAMES ARE THE TRADEMARKS OF THEIR RESPECTIVE OWNERS.

