2.4 Heapsort Demo

click to begin demo
Heapsort

Starting point. Array in arbitrary order.

we assume array entries are indexed 1 to N
Heap construction. Build max heap using bottom-up method.
Heap construction. Build max heap using bottom-up method.

sink 5
Heap construction. Build max heap using bottom-up method.

sink 5
Heap construction. Build max heap using bottom-up method.

sink 5
Heap construction. Build max heap using bottom-up method.

sink 4
Heap construction. Build max heap using bottom-up method.

sink 4
Heapsort

Heap construction. Build max heap using bottom-up method.

sink 3
Heapsort

Heap construction. Build max heap using bottom-up method.

sink 3
Heap construction. Build max heap using bottom-up method.

sink 3
Heapsort

Heap construction. Build max heap using bottom-up method.

sink 2

Diagram of a max heap with the root node labeled as 'S' and the diagram showing nodes labeled 'O', 'T', 'M', 'P', 'E', 'L', 'R', 'X', 'A'.
Heap construction. Build max heap using bottom-up method.

sink 2
Heap construction. Build max heap using bottom-up method.

sink 2
Heap construction. Build max heap using bottom-up method.

sink 2

7-node heap
Heap construction. Build max heap using bottom-up method.

sink 1
Heapsort

Heap construction. Build max heap using bottom-up method.

sink 1

```
X  T  S  P  L  R  A  M  O  E  E
1   3
```
Heapsort

Heap construction. Build max heap using bottom-up method.

end of construction phase
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 11
**Heapsort**

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 11
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

---

Heapsort is a sorting algorithm based on a binary heap data structure. It works by first building a max heap from the input data. Then, it repeatedly removes the maximum element (the root of the heap) and places it at the end of the list, then heapifies the remaining elements to maintain the max heap property. This process is repeated until the heap is empty, resulting in a sorted list.
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 10
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 10
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

**sink 1**
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

```
sink 1
```

```
S
 P
 O
 M
 E

P
 L

T

R
 E

X

S  P  R  O  L  E  A  M  E  T  X
1   3   6
```
Heapsort

Sortdown. Repeatedly delete the largest remaining item.
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 9
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 9
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Sortdown. Repeatedly delete the largest remaining item.
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

exchange 1 and 8

**Diagram:**

```
       R
      /|
     / | 1
   O   P
   / 
LO   E
   /     /
 8  M   E
   /     /
 S   T   A
```
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

-exchange 1 and 8-
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1

```
M
O P L
E E A
R S T X
```

```
M P E O L E A R S T X
1
```
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1
**Heap sort**

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1
Sortdown. Repeatedly delete the largest remaining item.
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 7
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

exchange 1 and 7
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1

```
R S T X
```

```
O A E M L E P R S T X
```

```
O 1
A 2
```

```
M L
```

```
O 1
A 2
```

```
E
```

```
P
```
**Heapsort**

*Sortdown.* Repeatedly delete the largest remaining item.

sink 1
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

exchange 1 and 6
**Sortdown.** Repeatedly delete the largest remaining item.

exchange 1 and 6
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.
Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 5
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 5
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

sink 1
Sortdown. Repeatedly delete the largest remaining item.
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.
Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 4
Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 4
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

**sink 1**
Heapsort

Sortdown. Repeatedly delete the largest remaining item.
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

exchange 1 and 3
**Heapsort**

**Sortdown.** Repeatedly delete the largest remaining item.

*exchange 1 and 3*
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

sink 1
Heapsort

Sortdown. Repeatedly delete the largest remaining item.
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 2
Heapsort

Sortdown. Repeatedly delete the largest remaining item.

exchange 1 and 2
Heapsort

Sortdown. Repeatedly delete the largest remaining item.
Heapsort

**Sortdown.** Repeatedly delete the largest remaining item.

**end of sortdown phase**

A

E E

L M O P

R S T X

A E E L M O P R S T X
Heapsort

**Ending point.** Array in sorted order.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>E</th>
<th>E</th>
<th>L</th>
<th>M</th>
<th>O</th>
<th>P</th>
<th>R</th>
<th>S</th>
<th>T</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td></td>
</tr>
</tbody>
</table>