

1 Sortowanie przez kopcowanie (heap sort).

```
Start
Dla  $i \leftarrow 1$  do długość  $values$  wykonuj
     $value \leftarrow values[i]$ 
     $child\_index \leftarrow i$ 
    Gdy  $child\_index > 0$  wykonuj
         $parent\_index \leftarrow (child\_index - 1)div2$ 
        Jeżeli  $values[parent\_index] \geq value$  to
            | Wyjdź z pętli
        koniec
         $values[child\_index] \leftarrow values[parent\_index]$ 
         $child\_index \leftarrow parent\_index$ 
    koniec
     $values[child\_index] \leftarrow value$ 
koniec
Dla  $i \leftarrow 0$  do długość  $values$  wykonuj
     $leaf\_index \leftarrow \text{długość } values - i - 1$ 
     $temp \leftarrow values[0]$ 
     $values[0] \leftarrow values[leaf\_index]$ 
     $values[leaf\_index] \leftarrow temp$ 
     $parent\_index \leftarrow 0$ 
    Gdy  $parent\_index * 2 + 1 < leaf\_index$  wykonuj
         $left\_index \leftarrow parent\_index * 2 + 1$ 
         $right\_index \leftarrow parent\_index * 2 + 2$ 
        Jeżeli  $right\_index < leaf\_index$  oraz  $values[right\_index] > values[left\_index]$ 
            to
            |  $max\_index \leftarrow right\_index$ 
        koniec
        w przeciwnym razie
        |  $max\_index \leftarrow left\_index$ 
    koniec
    Jeżeli  $values[parent\_index] \geq values[max\_index]$  to
        | Wyjdź z pętli
    koniec
     $temp \leftarrow values[parent\_index]$ 
     $values[parent\_index] \leftarrow values[max\_index]$ 
     $values[max\_index] \leftarrow temp$ 
     $parent\_index \leftarrow max\_index$ 
koniec
koniec
Wypisz listę  $values$ 
Stop
```