

Heap Sort

```
proc heapsort(A: array, n: array size)
  {Create heap}
  for r =  $\lfloor n/2 \rfloor$  downto 1 do
    sift(r,n,A[r])
  end for
  {Finish Sort}
  for m = n downto 2 do
    s  $\leftarrow$  A[m]
    A[m]  $\leftarrow$  A[1]
    sift(1,m-1,s)
  end for
end proc

proc sift(r: root, n: size of list, s: sift value)
  p  $\leftarrow$  r      {p: parent, c: child}
  while  $2*p \leq n$  do
    if  $2*p < n$  then
      if  $A[2*p] \geq A[2*p+1]$ 
        then c  $\leftarrow$   $2*p$ 
        else c  $\leftarrow$   $2*p+1$ 
      end if
    else
      c  $\leftarrow$   $2*p$ 
    end if
    if  $A[c] > s$  then
      A[p]  $\leftarrow$  A[c]; p  $\leftarrow$  c
    else
      exit while loop
    end if
  end while
  A[p]  $\leftarrow$  s
end proc
```