Caio Brighenti
COSC 302 - Analysis of Algorithms – Spring 2019
Lab 9

3. **Problem 6-5-3**
   We provide algorithms for each of the following procedures on a min-heap. A min-heap is a heap such that each node is always greater than or equal to its parent. These procedures work very similarly to the procedures for a max-heap, thus the pseudo-code provided is highly similar to that available in textbook 1 chapter 6.

```
HEAP-MINIMUM(A){
  return A[1]
}
```

```
HEAP-EXTRACT-MIN(A){
  if A.size < 1
    error "heap underflow"
  min = A[1]
  A[1] = A[A.size]
  A.size--
  MIN-HEAPIFY(A,1)
  return min
}
```

```
MIN-HEAPIFY(A,i){
  l = LEFT(i)
  r = RIGHT(i)
  if l ≤ A.size and A[l] < A[r]
    smallest = l
  else
    smallest = i
  if r ≤ A.size and A[r] < A[smallest]
    smallest = r
  if smallest ≠ i
    exchange A[i] with A[smallest]
    MIN-HEAPIFY(A,smallest)
}
```

```
HEAP-DECREASE-KEY(A,i,key){
  if key > A[i]
    error "new key is larger than current key"
  A[i] = key
  while i > 1 and A[PARENT(i)] > A[i]
    exchange A[i] with A[PARENT(i)]
    i = PARENT(i)
}
```

```
MIN-HEAP-INSERT(A,key){
  A.size++
  A[A.size] = ∞
  HEAP-DECREASE-KEY(A,A.size,key)
}
```