Loop Invariant

Max Heap:

Each node’s key is at least size of all descendant’s keys.

Suppose initially:

Indices of array A containing values (actually stored in Array)

Build - Max-Heap

for \( i = \lceil \text{A.length}/2 \rceil \) to 1

Max-Heapify(A, i)
Max Heapify(A, i)

Input: 

Output: 

What is good loop invariant for Build-Max Heap?

All indexes i+1, ..., n are roots of heaps

Initialization

All indexes \[\lfloor A.length / 2 \rfloor, ..., n\] are leaves, so roots of heaps

Maintenance

Since all larger indexes are roots of heap, the two children of i are roots of heaps, so satisfies conditions of Max Heapify, so Max Heapify makes a heap at i.

After, i is root of heap, so all children of i are roots of heaps, and all other indexes unchanged.

Termination

At end, index 1 is root of heap