AlgoExpert   Quad Layout   C++   12px   Sublime   Monokai   00:00:00

Prompt   Scratchpad   Our Solution(s)   Video Explanation   Run Code

Solution 1

```cpp
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

#include <vector>
using namespace std;

void buildMaxHeap(vector<int> &array);
void siftDown(int currentIdx, int endIdx, vector<int> &heap);

// Best: O(nlog(n)) time | O(1) space
// Average: O(nlog(n)) time | O(1) space
// Worst: O(nlog(n)) time | O(1) space
vector<int> heapSort(vector<int> array) {
  buildMaxHeap(array);
  for (int endIdx = array.size() - 1; endIdx > 0; endIdx--) {
    swap(array[0], array[endIdx]);
    siftDown(0, endIdx - 1, array);
  }
  return array;
}

void buildMaxHeap(vector<int> &array) {
  int firstParentIdx = (array.size() - 2) / 2;
  for (int currentIdx = firstParentIdx; currentIdx >= 0; currentIdx--) {
    siftDown(currentIdx, array.size() - 1, array);
  }
}

void siftDown(int currentIdx, int endIdx, vector<int> &heap) {
  int childOneIdx = currentIdx * 2 + 1;
  while (childOneIdx <= endIdx) {
    int childTwoIdx = currentIdx * 2 + 2 <= endIdx ? currentIdx * 2 + 2 : -1;
    int idxToSwap;
    if (childTwoIdx != -1 && heap.at(childTwoIdx) > heap.at(childOneIdx)) {
      idxToSwap = childTwoIdx;
    } else {
      idxToSwap = childOneIdx;
    }
    if (heap.at(idxToSwap) > heap.at(currentIdx)) {
      swap(heap[currentIdx], heap[idxToSwap]);
      currentIdx = idxToSwap;
      childOneIdx = currentIdx * 2 + 1;
    } else {
      return;
    }
  }
}
```