2) (10 pts) DSN (Heaps)

You are given an array, `arr`, of `size` integers. Write a **non-recursive** function that takes in this array and its size as input and returns 1 if the array represents a min-heap, and 0 otherwise. Recall that in an array implementation of a heap, index 0 isn't used and the root is stored at index 1. Since index 0 is unused, the function is checking if the array stores a valid heap in indexes 1 through `size-1`, inclusive, thus the number of nodes in the tree represented is `size-1`. **You may assume `size >= 2.`**

```
int isMinHeap(int arr[], int size) {

    for (int i = 1; 2*i < size; i++) {                  // 2 pts

        if (arr[i] > arr[2 * i])                        // 2 pts
            return 0;                                   // 1 pt

        if (2*i+1 < size && arr[i] > arr[2*i+1])        // 3 pts
            return 0;                                   // 1 pt

    }

    return 1;                                           // 1 pt

}
```

Grading note: if 1 is returned pre-maturely, take off 4 points (so max score of 6 in this case).