

# CSCI GW2

Brock Ellefson, Seth Severa, Yue Hou

September 13, 2017

- 1 Let  $A$  be an unsorted array of  $n$  integers, with  $A[0] \geq A[1]$  and  $A[n - 2] \leq A[n - 1]$ . Call an index  $i$  a local minimum if  $A[i]$  is less than or equal to its neighbors**

**a**

Check the middle element and its neighbors values. If the middle's value is smaller than its neighbors, it is the local minimum. If not, we call the function again passing half of the array containing the smaller element.

**b**

L:  $A[i]$  is a local min or  $A[i-1]$  or  $A[i+1]$

- 2 The `rand()` function in the standard C library returns a uniformly random number in  $[0, \text{RANDMAX} - 1]$ . Does `rand() mod n` generate a number uniformly distributed in  $[0, n-1]$ ?**

We cannot always generate a uniform number because there can be a case where  $n$  is not a uniformly divisible number.

- 3 In this class, we assume the real-RAM model of computation for our analysis. Explain why we must define what model of computation we are using in an algorithms class, or, more generally, when talking about the complexity of an algorithm.**

When talking about the complexity of an algorithm, different models can handle situations more or less efficiently. For instance, a major difference between turing machine and real-RAM machine is that a turing machine has an infinite tape, where as the real-RAM has an memory pool. This differences lead to different complexity on algorithms.

- 4 Which of the following correctly capture the runtime complexity of Mergesort**

b,e,h

Mergesort always runs in  $n \log n$  because it evaluates its partitions the same amount of times regardless of its content.