Analyzing the Running Time of a Simple Recursive Algorithm

A Suggested Exercise

About This Exercise

This exercise is intended to help you to practice and improve your skills in analyzing the running time of a simple recursive algorithm.

Problems To Be Discussed in the "Tutorial"

Recall the "Maximal Element in Part of an Integer Array" problem, considered in Reading #2. The following algorithm was proved to correctly solve this problem in the exercise for that reading.

```
maxInRange2 ( integer[] A, integer low, integer high ) {
1. if (low == high) {
2.   return A[low]
   } else {
3.   return max(maxInRange2(A, low, high − 1), A[high])
   }
}
```

1. Using the uniform cost criterion, write a ***recurrence*** for the number $T_{\text{max}}(k)$ of steps used by this algorithm when $0 \leq \texttt{low} \leq \texttt{high} \leq \texttt{A.length} - 1$ and $\texttt{high} - \texttt{low} + 1 = k \geq 1$.
2. Guess a ***solution*** for this recurrence — that is, guess an expression for $T_{\text{max}}(k)$ that is not in the form of a recurrence.
3. Prove that your guess is correct.