CSC236 winter 2020, quiz week 6_2

first/given name: **last/family name:**

utorid:

Use induction to prove the correctness of the function below, with respect to the precondition and postcondition stated in the docstring. (You do not need to restate or further formalize the given pre- and postcondition - though you may do so if you find it helpful.)

```
1 def isuniform(A):
2   """Pre: A is a list
3   Post: Return True if and only if every element in A is the same.
4   """
5   if len(A) <= 1:
6     return True
7   return A[0] == A[1] and isuniform(A[1:])
```

Solution Basis: for any list of length 0 or 1, we should return True according to the postcondition. And our function does this, by lines 5-6.

Inductive step: Let $n \in \mathbb{N}^+$, and assume that our function is correct on inputs of size n.

Let A be a list of length $n+1$. Because $n+1 \geq 2$, we reach line 7 in the code.

Case 1: Suppose A is uniform. It clearly follows that $A[0] = A[1]$ and that $A[1:]$ (indeed, any sublist of A) is uniform. So our function returns True, as required by the postcondition.

Case 2: A is not uniform. Then by definition, there exists an index i such that $A[0] \neq A[i]$ (it is easy to see that the negation of this statement entails that A uniformly consists of instances of $A[0]$). Let i' be the smallest such index. If $i' = 1$, then we return False, by the first condition of line 7. If $i' > 1$, then it follows that $A[1:]$ is not uniform, because the sublist contains an element not equal to $A[0]$, and the first element of the sublist ($A[1]$) is equal to $A[0]$. Thus, by the IH, because $A[1:]$ is not uniform, the recursive call returns False, meaning that we return False from the 'or' on line 7, as required.

In either case, our function matches the postcondition for an arbitrary input of size $n+1$, concluding our induction. ■