

Activity Sheet 3

Manager name:

Recorder name:

Speaker name:

Section 2.3

1. The following algorithm determines if an array of numbers is sorted:

```
// Input: A[0, ..., n-1]
// Output: true or false depending on whether the array is sorted
// Variables used: i — loop index
for i = 2, ..., n-1:
    if A[i-1] > A[i]:
        return false
return true
```

- a. The algorithm has one mistake in it. Correct it.
- b. Determine the input size and the basic operation for this algorithm.
- c. Compute the algorithm's time efficiency.

Section 2.4

2. The following algorithm determines the number of ones in the binary representation of a number n :

ALGORITHM ones(n)

// Input: n — a nonnegative integer

// Output: The number of ones in the binary representation of n

if $n \leq 1$:

 return n // n is 0 or 1

else:

 lastDigit = $n \bmod 2$ // lastDigit=1 if n is odd, lastDigit = 0 if n is even

 otherDigits = ones($n / 2$) // integer division here

 return lastDigit + otherDigits

- a. Demonstrate a run of this algorithm for $n = 11$, and confirm that it works in that instance.
- b. Determine the basic operation for this algorithm.
- c. Establish a recurrence relation for the running time of this algorithm.
- d. Use the method of backward substitutions on this recurrence to compute the running time.