CSE131 Quiz 2: Recursion
September 14, 2007

(int 1) int foo(int j, int k) {
(int 2) if (j == k)
(int 3) return 0;
(int 4) else if (j > k)
(int 5) return -1;
(int 6) else
(int 7) return 2 + foo(j+1, k-1);
(int 8) }

1. Using the substitution model, show the execution of foo(6, 10). Circle the return value.

\[
\begin{align*}
\text{foo}(6, 10) \\
2 + \text{foo}(7, 9) \\
2 + 2 + \text{foo}(8, 8) \\
2 + 2 + 0 \\
\end{align*}
\]

2. In the execution of foo(7, 8), draw the execution stack as it would look just before the first time line 5 is executed.

3. Recall that a % b == 0 when a is divisible by b.
Write a recursive method called numFactors with the following specification:

PARAMETERS: positive integers n and k
RETURN VALUE: the number of times k is a factor of n
EXAMPLES: numFactors(12, 7) is 0, because 7 is not a factor of 12.
numFactors(25, 5) is 2, because 25 = 5*5.
numFactors(52, 13) is 1, because 52 = 13*4.
numFactors(32, 4) is 2, because 32 = 4*4*2.
HINT: Think about what happens to the number of factors when you divide n by k.

\[
\begin{align*}
\text{int numFactors(int n, int k)} \{ \\
\text{if (n \% k != 0) // not divisible} \\
\text{return 0;} \\
\text{else} \\
\text{return 1 + numFactors(n/k, k);} \\
\}
\end{align*}
\]