Hw1 WriteUp

Test Case Execution for PAD:

CL-USER> (PAD 1)

1

CL-USER> (PAD 2)

1

CL-USER> (PAD 3)

2

CL-USER> (PAD 4)

2

CL-USER> (PAD 5)

3

CL-USER> (PAD 6)

4

CL-USER> (PAD 7)

5

CL-USER> (PAD 8)

7

CL-USER> (PAD 9)

9

CL-USER> (PAD 10)

12

CL-USER> (PAD 11)

16

CL-USER> (PAD 12)

21

Test Case Execution for SUMS:

CL-USER> (SUMS 0)

0

CL-USER> (SUMS 1)

0

CL-USER> (SUMS 2)

0

CL-USER> (SUMS 3)

1

CL-USER> (SUMS 4)

1

CL-USER> (SUMS 5)

2

CL-USER> (SUMS 6)

3

CL-USER> (SUMS 7)

4

CL-USER> (SUMS 8)

6

CL-USER> (SUMS 9)

8

CL-USER> (SUMS 10)

11

Test Case Execution for ANON:

CL-USER> (ANON '42)

?

CL-USER> (ANON 'FOO)

?

CL-USER> (ANON '(((L E) F) T))

(((? ?) ?) ?)

CL-USER> (ANON '(5 FOO 3.1 -0.2))

(? ? ? ?)

CL-USER> (ANON '(1 (FOO 3.1) -0.2))

(? (? ?) ?)

CL-USER> (ANON '(((1 2) (FOO 3.1)) (BAR -0.2)))

(((? ?) (? ?)) (? ?))

CL-USER> (ANON '(R (I (G (H T)))))

(? (? (? (? ?))))

Question 1) Small values of N for PAD are quickly evaluated. However, when we increase the value of N to something much larger, such as 200, PAD does not terminate. This is to be expected, because there are two recursive calls to PAD within the function, so the number of function calls increases exponentially with increasing values of N so that when there are extremely large values of N, the function becomes extremely slow. Even while I was writing this up, my function call with N equal to 200 still hasn’t terminated.

Question 2)

Below I have written up a chart, where the top line is a list of the Padovan numbers for N=0 through N=10, and the bottom line is a list of the corresponding number of additions executed by PAD for each of these Padovan numbers (what SUMS returns).

1 1 1 2 2 3 4 5 7 9 12

0 0 0 1 1 2 3 4 6 8 11

We know that for the first three Padovan numbers, no additions occur, because these are our base cases. Beyond this, however, we notice that the number of additions is equal to the number of PAD additions when N=N-2 plus the number of PAD additions when N=N-3 (these are our recursive calls) plus 1 for the addition of the two numbers returned by these recursive calls. Therefore, to find each value in the second row, we need only add together the value two to the left of it and three to the left of it, and add an additional 1.

In addition, we note that each of the values returned by SUMS are 1 less than the Padovan number returned with the same input for N in PAD. For example, SUMS(6) returns 3, which is one less than PAD(6), which returns 4.