CS 2233 Discrete Mathematical Structures – Fall 09

10/19/09

6. Homework

Due **10/28/09** before class

Please refer to the corresponding exercise sections in the textbook (Rosen, 6th edition).

Annotate all your proofs with comments/text in order to receive full credit.

4.5 (5 points) Use the loop invariant `(I): sum=a[0]+...+a[i]` to show that the code below correctly computes the sum of all elements in an array `a[0..n-1]` of `n` integers, for any `n`≥ 1. First, use induction to show that (I) is indeed a loop invariant, and then draw conclusions for the termination of the while loop.

```
int computeSum(int[] a[0..n-1]){
  sum = a[0];
  i=0;
  while(i<n-1){
    // (I) sum=a[0]+...+a[i]
    i++;
    sum = sum + a[i];
  }
  return sum;
}
```

4.3 (page 308)

(a) (4 points) 8 a, d. First write down the first six elements of the sequence, and then try to find a recursive solution. Do not forget the base case. (You do not need to prove the correctness of your solution.)

(b) (4 points) 18. Use strong induction. *(Hint: Matrix multiplication is explained on page 248. Use the recursive definition of the Fibonacci numbers on page 297.)*

(c) (5 points) 26a (only for the first three applications of the recursive definition), 26c. *(Hint: $a|b$ reads a divides b. $a|b \leftrightarrow \exists k \in \mathbb{Z} : b = a * k$. For example* $5|15$ *and* $7|28$. *See page 201.)*

(d) (4 points) Give a **recursive** algorithm to compute the maximum element of an array of n integers. Also give the initial call to your recursive algorithm.

7.1 (page 457)
(4 points) 8a,b. Use the expansion method to find explicit formulas. You do **not** need to prove the correctness of your solutions.