

CS200 - Problem Set 1

Due: Monday, Sep. 18. Upload to Canvas before the beginning of class

Please read the sections of the syllabus on problem sets and honor code before starting this homework.

1. *Errors in Inductive Proofs*

- (a) **[2 points]** [DM 2.5.14](#)
- (b) **[6 points]** Consider bit sequences in which 1s do not appear consecutively, except in the rightmost two positions. For instance, 0010100 and 1000011 are bit sequences of this sort, and 0011000 is not.

What is wrong with the following “proof” that there are 2^n “allowed” sequences of length n ?

“Proof” by induction: Let s_n be the number of n -bit sequences in which 1’s do not appear consecutively except in the rightmost two positions. Let $P(n)$ be the predicate that $s_n = 2^n$. We will prove by induction on n that $P(n)$ is true for all $n \geq 1$.

Base Case: $P(1)$ asserts that $s_1 = 2$. This assertion is clearly correct: the only possible sequences of length one are 0 and 1, and both 0 and 1 are allowed.

Inductive Case: Let $k \geq 1$, and assume for induction that $P(k)$ is true. That is, we assume that $s_k = 2^k$. We will prove that $P(k + 1)$ is true. For every bit sequence of length k , we can append either 0 or 1 at the right end. In the case that we append 0, we always end up with a valid sequence. In the case that we append 1, we may create 11 in the last two positions, but that’s allowed. Therefore,

$$s_{k+1} = 2s_k = 2 \cdot 2^k = 2^{k+1},$$

and so $P(k + 1)$ is true.

Thus, by induction, $P(n)$ is true for all $n \geq 1$.

2. *Inductive Proofs*

- (a) **[11 points]** Prove using induction that for $n \geq 0$, $7^n - 2^n$ is divisible by 5. (An integer m is divisible by an integer r if $m = r \cdot g$, where g is some other integer.)
- (b) **[11 points]** Prove using induction that $2^n > n^2$ whenever n is an integer greater than 4.

- (c) [11 points] Prove using induction that Algorithm 1 correctly multiplies a non-negative n and an integer b . Hint: define $P(n)$ to be the predicate: `Mult(n,b)` correctly outputs the product of n and b .

Algorithm 1: `Mult(n, b)`

```
Input  : Non-negative integer  $n$ , and integer  $b$ 
Output:  $n \times b$ 
/* Base Case
1 if  $n == 0$  then
2   |   return 0
3   |
4 else
5   |   // Recursive step
6   |   return  $b + \text{Mult}(n - 1, b)$ 
7 end
```

*/

- (d) [11 points] For your programming assignment this week, you have to implement a recursive algorithm. Use an inductive proof to prove that your algorithm works correctly.

3. How long did you spend on this homework?