CPSC 320: Intermediate Algorithm Design 2019/20 Term 2

Tutorial 1

Prof. Nick Harvey *University of British Columbia*

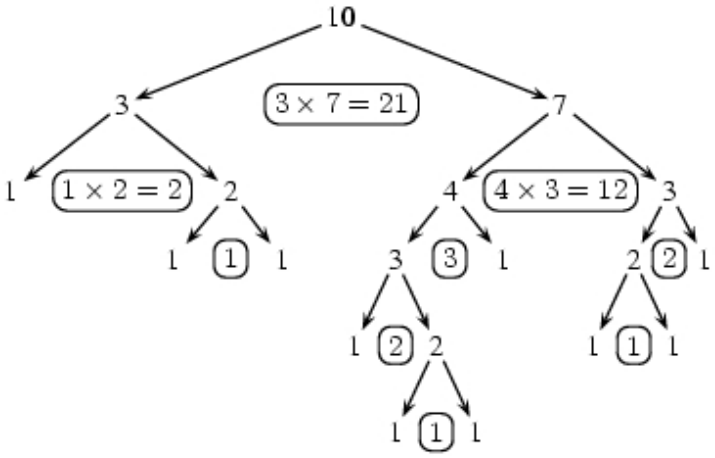
1. (Review of proof by contradiction) Let a, b and n be non-negative real numbers. Suppose that $a \cdot b = n$. Prove that the statement

$$\text{either} \quad a \leq \sqrt{n} \tag{1}$$
$$\text{or} \quad b \leq \sqrt{n} \tag{2}$$

   is true.

2. (Review of proof by induction) Suppose you begin with a pile of n stones, and split this pile into n piles of one stone each by successively splitting a pile of stones into two smaller piles. Each time you split a pile you multiply the number of stones in each of the two smaller piles you form, so that if these piles have r and s stones in them, respectively, you compute $r \cdot s$. Show that no matter how you split the pile, the sum of the products computed at each step equals $n(n-1)/2$.

   Example: Here is an example that shows how the computation proceeds:



   The sum is $21 + 2 + 12 + 1 + 3 + 2 + 2 + 1 + 1 = 45$, which is indeed $10 \cdot 9/2$.

3. (Review of loop invariants) The code snippet of the binary search is shown below. Prove that the code is correct in the case when *target* is an element of array A and array A is sorted in increasing order.

```
function BinarySearch(A, target)
    lo ← 1 (assume array starts at index 1)
    hi ← len(A)
    while lo ≤ hi do
        mid ← (lo + hi)/2
        if A[mid] = target then
            return mid
        else if A[mid] < target then
            lo ← mid + 1
        else
            hi ← mid − 1
        end if
    end while
    return "not found"
end function
```