

## CS 362, HW 5

Prof. Jared Saia, University of New Mexico

1. Prove that any tree with  $n$  nodes has  $n - 1$  edges. Hint: to get a smaller subproblem, remove a leaf node, i.e. a degree 1 node, from the tree. To prove that any tree has a leaf node, note that trees do not have cycles and consider a path that starts at the root node.
2. A product is sold in bags of sizes 3, 5 and 8 kilos. Prove that given an unlimited supply of bags of these sizes, you can fill an order for exactly  $n$  kilos for any  $n \geq 8$ . Hint: Be careful about the base cases, and also the lower bound for your IH.
3. Jesse wants to know the minimum number of bags needed to fill an order of size  $n$  kilos assuming unlimited supply of bags of sizes 3, 5 and 8 kilos. Let  $f(n)$  be the minimum number of bags needed to fill an order of size  $n$  if this is possible, or  $\infty$  if an order of size  $n$  can not be filled.
  - (a) Write a recurrence relation for  $f(n)$ . Hint: Use the minimum function in your recurrence relation; it may also be useful to use the value  $\infty$  for some of your base cases.
  - (b) Use your recurrence relation to fill in an array of  $f(n)$  values for  $n \in [0, 16]$ .
  - (c) Describe an algorithm to compute  $f(n)$  for any given  $n$  using your recurrence relation. What is the runtime of your algorithm? How might you augment your algorithm to return the actual set of bags used to achieve the minimum value  $f(n)$ ? Hint: you can put back-arrows in your array.

4. Consider the following function:

```
int f (int n){
    if (n==0) return 2;
    else if (n==1) return 5;
    else{
        int val = 2*f (n-1);
        val = val - f (n-2);
        return val;}}
```

- (a) Write a recurrence relation for the *value* returned by  $f$ . Solve the recurrence exactly. (Don't forget to check it)
- (b) Write a recurrence relation for the *running time* of  $f$ . Get a tight asymptotic bound (i.e.  $\Theta$ ) on the solution to this recurrence.