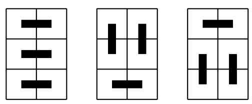
Problem Set 2

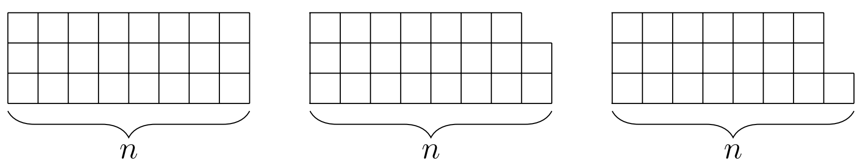
CS 331H

Due Tuesday, February 28

1. How many ways can one tile a $3 \times n$ rectangle using 2×1 tiles? For example, there are 3 ways to tile a 3×2 rectangle:



   (a) [50%] Show how to compute the answer in $O(n)$ time, assuming that the word size can represent numbers as large as the answer. You may find it useful to consider the number of ways to tile all of the following figures:



   (b) [15%] Repeat part (a) for tiling a $k \times n$ rectangle using 2×1 rectangles, for any constant k. How does your complexity scale with k?

   (c) [15%] By expressing the recursion in terms of matrices, show how to compute the part (a) answer in $O(\log n)$ time, again assuming the word size can represent numbers as large as the answer.

   (d) [20%] Now suppose that the word size is $\Theta(\log n)$. How much time do your solutions take?

2. You are given a sequence of n integers, $x_1, \ldots, x_n$, and an integer $k \in [n]$. Each integer is polynomially large. Find the contiguous subset of size at least k with maximum *average*. That is, find two indices $s, t \in [n]$ with $t \geq s + k - 1$ that maximize

$$\frac{1}{t - s + 1} \sum_{i=s}^{t} x_i.$$

Full credit requires $O(n \log n)$ time. **Hints:** Can you find the length $\geq k$ contiguous subset of maximum *sum* in $O(n)$ time? Then, can you determine *whether the maximum average is nonnegative* in $O(n)$ time? Can you then search over the space of possible averages, to see which ones are possible?

3. Consider a weighted, directed graph where all distances lie in $[1, 2)$.

   (a) Show that a variant of BFS computes single source shortest paths in $O(m)$ time.

   Hint: Visit all the nodes in non-decreasing order of floor(distance(u)). That is, after visiting the source s, visit all the nodes with distance in [1,2) in some order, then all the nodes with distance in [2,3) in some order, etc.

   (b) Extend your result to $O(mC)$ time and $O(m)$ space for distances in $[1, C)$ for any $C \geq 1$.