COURSE REVIEW

Recitation week 11

DYNAMIC PROGRAMMING – MAX SUM BITONIC SUBARRAY

- Consider an array of length *n*. We call a subarray bitonic if it can be split into 2 subarrays, the first increasing and the second decreasing (one or both may be empty). For example: A = [1, 2, 0, 2, 3, 2, 1, 8]
- Using linear time and auxiliary memory, given some array *A*, find the bitonic subarray *A'* of *A* with maximum sum.
- We can think of this problem as an intersection of the maximum sum increasing subarray and maximum sum decreasing subarray.
- In other words, we want to store the maximum increasing array ending at i and the maximum decreasing array beginning at i
  - $MaxSumIncreasing[i] = A[i] + I\{A[i] > A[i-1]\} * MaxSumIncreasing[i-1]$
  - $MaxSumDecreasing[i] = A[i] + I\{A[i] < A[i+1]\} * MaxSumDecreasing[i+1]$
  - $MaxSumBitonic[i] = MaxSumIncreasing[i] + MaxSumDecreasing[i] - A[i]$

RANDOMIZED ALGORITHMS AND PROBABILISTIC ANALYSIS

- The vast majority of the problems we explored in randomized algorithms (and probabilistic analysis) involved the selection of random (indicator) variables.
- An indicator random variable is a binary (i.e., 0 or 1 are its possible values) random variable which "tracks an event". In other words, an indicator random variable $X = I\{E\}$ for an event E is 1 if E happens and 0 otherwise.
- Therefore, the expected value of an IRV is the probability that the event it tracks occurs.
- We used these random variables to calculate the expected running time of an algorithm or number of some "thing" occurring.
- We do this since we don't know what events are going to happen, but *some of them will happen*. By analyzing the probability that these events happen and the resulting value conditioned on this fact, we can get the expected total (running time or desired value)

RANDOMIZED ALGORITHMS AND PROBABILISTIC ANALYSIS – "THE CARS PROBLEM"

- Let's review the "car clumps" problem
- Underscores the fundamental notion that solving the problem is equivalent* to finding good indicator random variables
- *the rest is just algebra

[16 pts] Suppose there are n cars, each of which has a different, but constant, speed (it does not change over time). We choose an order of these cars uniformly at random and let them go along an infinitely long one-lane tunnel. They cannot pass each other so if a faster car ends up behind a slower car, it must slow down to the speed of the slower car. Eventually the cars will clump up in traffic jams. Find the expected number of clumps of cars. (A clump contains one or more cars.)

- The first step is to fully comprehend what is being requested (what should we track)?
- Cognizant of this, we can consider how to model this as indicator random variables – the number of clumps is the same as the number of cars leading a clump. So if X is the number of clumps, $X = \sum_i X_i$ where $X_i = I\{car\ i\ leads\ a\ clump\}$

RANDOMIZED ALGORITHMS AND PROBABILISTIC ANALYSIS – "THE CARS PROBLEM"

[16 pts] Suppose there are n cars, each of which has a different, but constant, speed (it does not change over time). We choose an order of these cars uniformly at random and let them go along an infinitely long one-lane tunnel. They cannot pass each other so if a faster car ends up behind a slower car, it must slow down to the speed of the slower car. Eventually the cars will clump up in traffic jams. Find the expected number of clumps of cars. (A clump contains one or more cars.)

- $X = \sum_i X_i$ where $X_i = I\{car\ i\ leads\ a\ clump\}$
- We want $E[X]$ so we know to apply linearity of expectation
- $E[X] = E[\sum_i X_i] = \sum_i E[X_i]$
- Now it just remains to calculate the value of X_i. For this, we consult the problem again to find the distribution of X_i.
- Since each car has a different, but constant, speed, we know that a car leads a clump if and only if it is slower than everything in front of it. If there are i cars (including the one we are considering), this occurs with probability $\frac{1}{i}$
- From this we just take the sum $\sum_{i=1..n} \frac{1}{i} = \Theta(\log n)$

TAKING A STEP BACK

- This problem illustrates some interesting notions that can be useful in general for solving randomized/probabilistic problems.
- We did not need to know the underlying distribution until the very end. The only thing we needed to know was that we could calculate the expected value.
- There are three things to keep in mind, then, when tackling these problems:
    1. Where does randomness occur? What are the events?
    2. Can I calculate the probabilities of my selected IRV?
    3. Can I combine my IRV to find the desired result?