CS 4349.501 Homework 11

Due Wednesday December 6th, in class

November 27, 2017

Please answer all questions. There are four regular questions and one extra credit question. Each student must write their solutions in their own words and submit their solutions on paper at the beginning of class. Include your name and/or Net ID at the top of each page. And please staple your papers together.

This assignment is due before class begins Wednesday.

1. Consider the following modification to the SELECT algorithm from early in the semester, which partitions the input array into $\lceil n/6 \rceil$ blocks of size 6, instead of $\lceil n/5 \rceil$ blocks of size 5, but is otherwise identical.

```
MOM₆SELECT(A[1 ... n], k):
  if n ≤ 36
      use brute force
  else
      m ← ⌈n/6⌉
      for i ← 1 to m
          M[i] ← MEDIANOF6(A[6(i−1)+1 ... 6i])
      mom₆ ← MOM₆SELECT(M[1 ... m], ⌊m/2⌋)

      r ← PARTITION(A[1 ... n], mom₆)

      if k < r
          return MOM₆SELECT(A[1 ... r−1], k)
      else if k > r
          return MOM₆SELECT(A[r+1 ... n], k−r)
      else
          return mom₆
```

(a) In lecture, we saw the worst-case running time of the original SELECT algorithm follows the recurrence $T(n) \le O(n) + T(n/5) + T(7n/10)$. State a similar recurrence for the running time of $\text{Mom}_6\text{Select}$, assuming that MEDIANOF6 runs in $O(1)$ time.

(b) What is the running time of $\text{Mom}_6\text{Select}$ obtained by solving your recurrence?

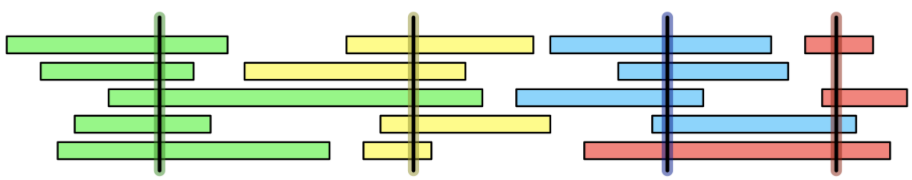
2. In the BACKPACK problem, you are given a collection of books of different costs and are tasked with storing as expensive a subset of books as possible so you can sell them at the bookstore in only one trip. Unfortunately, the books are very heavy, and you are limited in how much total weight you can carry.

   Formally, you are given two arrays $C[1 .. n]$ and $W[1 .. n]$ of positive integers where $C[i]$ is the cost of book i in dollars and $W[i]$ is the weight of book i in pounds. You are also given a positive integer M which is the maximum load you can carry in pounds. Your goal is to compute the maximum total cost of any subset of books you can carry from books 1 through n.

   (a) For any integers i and m with $0 \leq i \leq n$ and $0 \leq m \leq M$, let $MaxCost(i, m)$ be the maximum cost of any subset of books 1 through $\boldsymbol{i}$ that have a total weight of at most m. Give a recurrence definition for $MaxCost(i, m)$. Don't forget the base cases!

   (b) Design and analyze a dynamic programming algorithm for solving the BACKPACK problem based on your recurrence from part (a). You should get a running time of $O(nM)$.

3. Let X be a set of n intervals on the real line. We say that a set P *stabs* X if every interval in X contains at least one point in P.

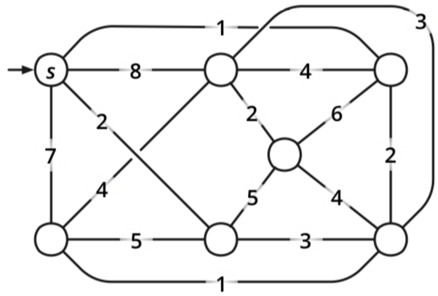


A set of intervals stabbed by four points (shown here as vertical segments).

   Consider the following greedy strategy for computing a smallest set P of points that stabs X: Let x be the interval whose *right* endpoint comes furthest to the *left*. We add the right endpoint of x to P, remove all intervals containing p from X, and recursively add a smallest stabbing set on the remaining intervals to P.

   Prove that this strategy does compute a smallest stabbing set for X.

4. Consider the weighted graph pictured below.



(a) Draw a depth-first spanning tree rooted at s.

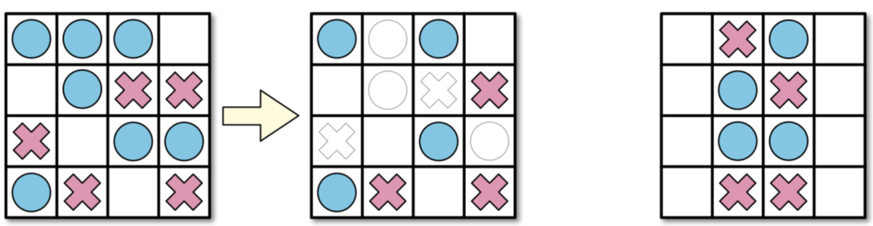
(b) Draw a breadth-first spanning tree rooted at s.

(c) Draw a shortest-path tree rooted at s.

(d) Draw a minimum spanning tree.

Some of these subproblems may have more than one correct answer.

5. **Extra credit** (worth 1 full question): Consider the following solitaire game. The puzzle consists of an $n \times m$ grid of squares, where each square may be empty, occupied by a red stone, or occupied by a blue stone. The goal of the puzzle is to remove some of the given stones so that the remaining stones satisfy two conditions: (1) every row contains at least one stone, and (2) no column contains stones of both colors. For some initial configurations of stones, reaching this goal is impossible.



A solvable puzzle and one of its many solutions. An unsolvable puzzle.

Prove that it is NP-hard to determine, given an initial configuration of red and blue stones, whether the puzzle can be solved. *[Hint: 3SAT. Be sure to do the reduction in the correct direction and prove it correct!]*