Assignment 4 (due Friday, March 22nd, 6:00pm)

Instructions:

• Hand in your assignment using Crowdmark. Detailed instructions are on the course website.
• Give complete legible solutions to all questions.
• Your answers will be marked for clarity as well as correctness.
• For any algorithm you present, you should justify its correctness (if it is not obvious) and analyze the complexity.

1. [20 marks] Suppose we have a set $X$ of $n$ pairs of numbers $(a_i, b_i)$, such that for each $i$ the sum of $a_i + b_i = m$. You can assume all $a_i$ and $b_i$ are non-negative integers. Our goal is to find out if it is possible to divide $X$ into two disjoint subsets of size $n/2$, $S$ and $R = X - S$, such that the average of the $a$ values in both $S$ and $R$ are strictly greater than $m/2$ (or equivalently the sum of the $a$ values is greater than $mn/4$).

**Example.** Consider the following example with $n = 4$ pairs of numbers where $m = 200$.

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_1$</td>
<td>110</td>
<td>90</td>
</tr>
<tr>
<td>$p_2$</td>
<td>84</td>
<td>116</td>
</tr>
<tr>
<td>$p_3$</td>
<td>120</td>
<td>80</td>
</tr>
<tr>
<td>$p_4$</td>
<td>94</td>
<td>106</td>
</tr>
</tbody>
</table>

The answer to this question is YES because if we set $S = \{p_1, p_4\}$ and $R = \{p_2, p_3\}$, then the $a$ values in $S$ is $204 > 200$. Similarly the $a$ values in $R$ is $204 > 200$.

(a) [10 marks] Define the following subproblems ($i = 0, \ldots, n$, $j = 0, \ldots, n$, $\ell = 0, \ldots, mn$):

$$C[i, j, \ell] = \begin{cases} 1 & \text{if there exists a subset } T \subseteq \{1, \ldots, i\} \text{ such that } |T| = j \text{ and } \sum_{k \in T} a_k = \ell \\ 0 & \text{else} \end{cases}$$

First give a dynamic programming algorithm to compute $C[i, j, \ell]$ for all $i, j, \ell$. Remember to derive the base cases and a recurrence formula, provide justification for your formula, and write and analyze your pseudocode. The running time and space should be polynomial in $n$ and $m$.

(b) [5 marks] Now using the table $C[\cdot, \cdot, \cdot]$ computed in part (a), solve the problem from part (a), i.e., decide whether there exists a subset $S \subseteq \{1, \ldots, n\}$ of size $n/2$ such that the average of $\{a_k : k \in S\}$ is more than $m/2$ and the average of $\{a_k : k \in R\}$ is also more than $m/2$. The running time and space should be polynomial in $n$ and $m$. 

1
(c) [5 marks] Write pseudocode to recover a subset $S$ with the stated property in (b) if it exists, and analyze the running time.

2. [20 marks] Given a sequence of $n$ integers $a_1, \ldots, a_n$ and an integer $t$, we want to divide the sequence into $t$ chunks so as to minimize the sum of the squares of the chunk sums. In other words, we want to find $1 < i_1 < i_2 < \ldots < i_{t-1} < n$ to minimize $(a_1 + \ldots + a_{i_1-1})^2 + (a_{i_1} + \ldots + a_{i_2-1})^2 + \ldots + (a_{i_{t-1}} + \ldots + a_n)^2$. Present an $O(tn^2)$-time dynamic programming algorithm to solve the problem. Give pseudocode to compute the minimum value.

[Hint: define $O(nt)$ subproblems. . .]

3. [20 marks] Given a directed graph $G(V, E)$ with $k$ strongly connected components, describe a $O((n + m)k)$-time strongly connected components algorithm that only uses BFS traversals. Solutions that use DFS will not get any credit.

[Hint: Think of the definition of what an SCC is and how to find the SCC of a particular vertex $v$. Your solution might require modifying the graph $G$ along the way.]

4. [20 marks] Given $n$ intervals $[a_1, b_1], \ldots, [a_n, b_n]$ where each interval $[a_i, b_i]$ has weight $w_i$, we want to find a subset of intervals whose union covers $[0, 1]$ while minimizing the total weight. Example: if $[a_1, b_1] = [-0.2, 0.6], w_1 = 1, [a_2, b_2] = [-0.1, 1.1], w_2 = 3, [a_3, b_3] = [0.5, 1.2], w_3 = 1$, then we would pick the first and third intervals, with total weight 2. [Note: this weighted problem does not appear to be solvable by a greedy approach.]

Show that this problem can be reduced to finding the shortest path between two fixed vertices in a certain weighted directed acyclic graph (DAG) with $O(n)$ vertices and $O(n^2)$ edges. Hence, design a $O(n^2)$ time reduction algorithm that uses as a subroutine an algorithm from class.

5. [20 marks] Consider a connected graph $G = (V, E)$ with $n$ vertices and positive edge weights $w_e > 0$ on the edges $e \in E$. Assume that the $w_e$’s are distinct in $G$. Let $T = (V, E')$ be a spanning tree of $G$. The bottleneck edge of $T$ is the edge in $E'$ with the highest edge weight. A spanning tree $T_b$ of $G$ is a minimum bottleneck spanning (MBST) tree of $G$ is there is no other spanning tree $T'$ of $G$ with a smaller bottleneck.

(a) [14 marks] Prove that every minimum spanning tree (MST) of $G$ is also an MBST.

(b) [6 marks] Give a counterexample to the claim that an MBST is always an MST.