Advanced Analysis of Algorithms - Midterm (Solutions)

L. Kovalchick
LCSEE,
West Virginia University,
Morgantown, WV
{lynn@csee.wvu.edu}

1 Problems

1. `Summation:` Obtain asymptotic tight bounds (upper and lower) on

$$\sum_{k=1}^{n} k^2 \log k.$$

Solution:
Observe that $\sum_{k=1}^{n} k^2 \log k$ is a monotonically increasing function; therefore, we can use approximation by integrals.
Lower Bound:
Note that $\sum_{k=1}^{n} k^2 \log k = 0 + \sum_{k=2}^{n} k^2 \log k$, since $\log 1 = 0$. Using our formula for approximating the lower bound of a monotonically increasing function by integrals, we have $\int_1^n k^2 \cdot \log k \;\; dk \leq \sum_{k=2}^{n} k^2 \log k$. Using integration by parts, we let $dv = k^2 \;\; dk \Rightarrow v = \int k^2 \;\; dk = \frac{k^3}{3}$ and $u = \log k \Rightarrow du = \frac{1}{k} \;\; dk$. Performing the integration, we have the following.

$$\begin{aligned}
\int k^2 \cdot \log k \;\; dk &= \frac{k^3}{3} \cdot \log k - \int \frac{k^3}{3} \cdot \frac{1}{k} \;\; dk \\
&= \frac{k^3}{3} \cdot \log k - \frac{1}{3} \int k^2 \;\; dk \\
&= \frac{k^3}{3} \cdot \log k - \frac{k^3}{9} + C
\end{aligned}$$

Using this antiderivative, we can evaluate the definite integral as follows.

$$\begin{aligned}
\int_1^n k^2 \cdot \log k \;\; dk &= \left[\frac{k^3}{3} \cdot \log k - \frac{k^3}{9} \right]_1^n \\
&= \frac{n^3}{3} \cdot \log n - \frac{n^3}{9} - \frac{1^3}{3} \cdot \log 1 + \frac{1^3}{9} \\
&= \frac{n^3}{3} \cdot \log n - \frac{n^3}{9} + \frac{1}{9}
\end{aligned}$$

Thus, we have $\sum_{k=1}^{n} k^2 \log k \in \Omega(n^3 \cdot \log n)$.
Upper Bound:
Using our formula for approximating the upper bound of a monotonically increasing function by integrals,

we have $\sum_{k=1}^{n} k^2 \log k \leq \int_1^{n+1} k^2 \cdot \log k \;\; dk$. Using the antiderivative obtained for the lower bound, we can evaluate the definite integral as follows.

$$\begin{aligned}
&= \left[\frac{k^3}{3} \cdot \log k - \frac{k^3}{9}\right]_1^{n+1} \\
&= \frac{(n+1)^3}{3} \cdot \log(n+1) - \frac{(n+1)^3}{9} - \frac{1^3}{3} \cdot \log 1 + \frac{1^3}{9} \\
&= \frac{(n+1)^3}{3} \cdot \log(n+1) - \frac{(n+1)^3}{9} + \frac{1}{9}
\end{aligned}$$

Thus, we have $\sum_{k=1}^{n} k^2 \log k \in O(n^3 \cdot \log n)$.
Observe that $\sum_{k=1}^{n} k^2 \log k$ is bounded below by $\Omega(n^3 \cdot \log n)$ and it is bounded above by $O(n^3 \cdot \log n)$. Therefore, $\sum_{k=1}^{n} k^2 \log k \in \Theta(n^3 \cdot \log n)$.
Note: For an alternative solution to this problem, please see Appendix (A). □

2. `Convexity:`

   (a) Let S_1 and S_2 be 2 convex sets. Argue that $S_1 \cap S_2$ is a convex set. (3 points.)

   Proof:
   Let $S = S_1 \cap S_2$. Now, consider any 2 elements in S, namely x and y and the parametric point $z = \lambda \cdot x + (1-\lambda) \cdot y,\ 0 \leq \lambda \leq 1$, which represents the straight line joining x and y. Since S_1 is convex, $z \in S_1$; likewise, since S_2 is convex, $z \in S_2$. Since $z \in S_1$ and $z \in S_2$, it follows that $z \in S_1 \cap S_2$. In other words, the set S is convex. □

   (b) What can you say about the function $f(x) = \sin x$ in the interval $[0, \pi]$, as regards convexity? Justify your answer mathematically (3 points)

   Solution:
   Observe that $f'(x) = \cos x$ and $f''(x) = -\sin x$. From first principles, we know that a function $f(x)$ is convex over a domain D, if and only if $f''(x) \geq 0,\ \forall x \in D$. In our case, the domain $D = [0, \pi]$ and $f''(\frac{\pi}{2}) = -\sin\frac{\pi}{2} = -1$, from which we can conclude that $f(x) = \sin x$ is not convex. In fact, $f(x)$ is a concave function, in the interval $[0, \pi]$.

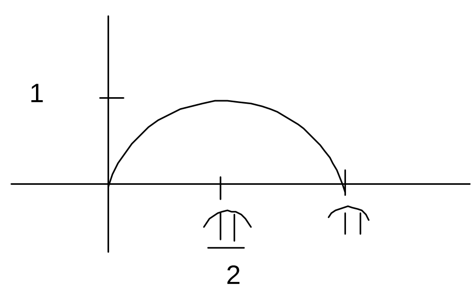

Figure 1: Graph of $f(x) = \sin x$ in the interval $[0, \pi]$.

□

3. `Probability:` Consider a bin containing 5 red balls and 7 black balls. What is the probability of obtaining 2 red balls in a single draw of 2 balls, where a draw of 2 balls is defined as:

   (a) One ball is drawn from the bin and then the second one is drawn, without replacing the first ball. (2 points)

   Solution:
   Let A be the event that the ball obtained in the first draw is red and B be the event that the ball obtained in the second draw is red. Then, $A \cap B$ is the event that A occurs, and then B occurs after A has occurred. Observe that the probability of obtaining a red ball on the first draw is $\mathbf{Pr}\{A\} = \frac{5}{12}$. Then, the probability of obtaining a second red ball from the remaining balls in the bin is $\mathbf{Pr}\{B|A\} = \frac{4}{11}$. Thus, we have the following.

$$\begin{aligned}\mathbf{Pr}\{obtaining\ 2\ red\ balls\} &= \mathbf{Pr}\{A \cap B\} \\ &= \mathbf{Pr}\{A\} \cdot \mathbf{Pr}\{B|A\} \\ &= \frac{5}{12} \cdot \frac{4}{11} \\ &= \frac{5}{3} \cdot \frac{1}{11} \\ &= \frac{5}{33}\end{aligned}$$

   □

   (b) One ball is drawn from the bin and the second one is drawn, after replacing the ball drawn first in the bin. (2 points)

   Solution:
   Let A be the event that the ball obtained in the first draw is red and B be the event that the ball obtained in the second draw is red. Then, $A \cap B$ is the event that A occurs, and then B occurs after A has occurred. Observe that since the first ball is replaced, our sample space for both the first and second draw consists of 12 balls (i.e., 5 red balls and 7 black balls). Thus, the probability of obtaining a red ball on the first draw is $\mathbf{Pr}\{A\} = \frac{5}{12}$. Then, the probability of obtaining a second red ball is $\mathbf{Pr}\{B|A\} = \frac{5}{12}$. Thus, we have the following.

$$\begin{aligned}\mathbf{Pr}\{obtaining\ 2\ red\ balls\} &= \mathbf{Pr}\{A \cap B\} \\ &= \mathbf{Pr}\{A\} \cdot \mathbf{Pr}\{B|A\} \\ &= \frac{5}{12} \cdot \frac{5}{12} \\ &= \frac{25}{144}\end{aligned}$$

   □

   (c) The two balls are *selected* at once from the bin. (2 points)

   Solution:
   Observe that the number of ways that we can choose 2 red balls from the 5 red balls that are in the bin is $\binom{5}{2}$, and the number of ways that we can choose 2 balls from the total number of balls in the bin is $\binom{12}{2}$. Now, the probability of choosing 2 red balls from the 12 balls in the bin is:

$$\begin{aligned}\frac{\binom{5}{2}}{\binom{12}{2}} &= \frac{\frac{5!}{2!\cdot 3!}}{\frac{12!}{2!\cdot 10!}} \\ &= \frac{5! \cdot 2! \cdot 10!}{2! \cdot 3! \cdot 12!}\end{aligned}$$

$$
\begin{aligned}
&= \frac{5! \cdot 10!}{3! \cdot 12!} \\
&= \frac{5 \cdot 4 \cdot 3! \cdot 10!}{3! \cdot 10! \cdot 11 \cdot 12} \\
&= \frac{5 \cdot 4}{11 \cdot 12} \\
&= \frac{5}{11 \cdot 3} \\
&= \frac{5}{33}
\end{aligned}
$$

□

4. `Search Trees:` Argue that if a node in a binary search tree has 2 children, then its successor has no left child, while its predecessor has no right child.

   Proof: We shall show that if a node x in a binary search tree T has 2 children, then its successor has no left child. The proof showing that its predecessor has no right child can be developed similarly. Without loss of generality, we assume that all keys in the tree T are distinct.

   Lemma 1.1 *If node x has a right child y, then its successor must be located in the subtree rooted at y.*

   Proof: Consider the case, in which x is the root r of T; in this case all nodes having keys greater than $key[x]$ and hence the successor of x are located in the subtree rooted at y. It follows that the lemma is trivially true. (See Figure (2).)

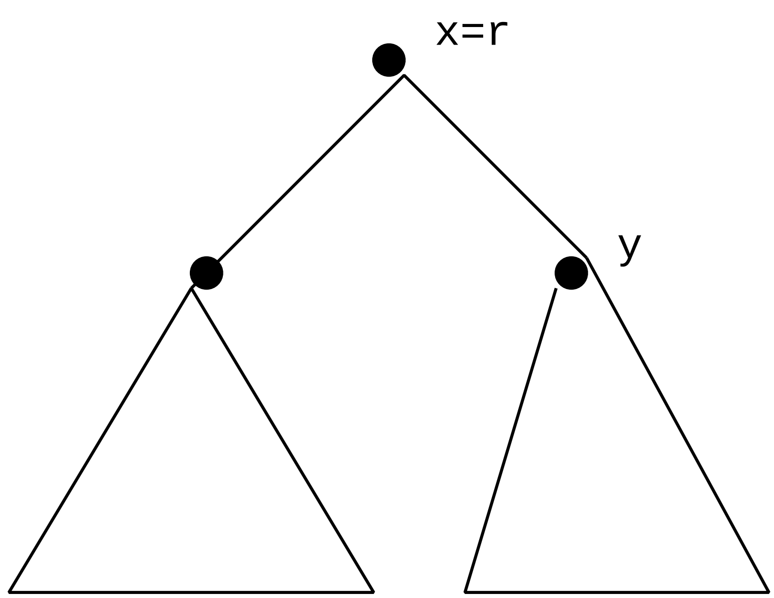

Figure 2: Simple case for successor of x

   Let x be a node in T, distinct from the root r of T and consider the *unique* path P from r to x in T. Unlike the case in which x was the root, a node having key value greater than $key[x]$ is not necessarily located at the subtree rooted at y. Let S denote the set of all nodes in the subtree rooted at y. We consider the following 2 cases:

(a) Node x is in the subtree rooted at the left child of r - Observe that the $key[r]$ and the keys of all the nodes in its right subtree are greater than the keys of any node in $S \cup \{x\}$ and therefore cannot be candidates for the successor of x. This property continues along the path P, till such time as an ancestor of x, viz. z, is reached, such that x is in the right subtree of z. Now, by the property of binary search trees, $key[z]$ and the keys of all the nodes in its left subtree are smaller than the key of any node in $S \cup \{x\}$; thus they can also be excluded from the candidate list of possible successors of x. The above two observations are recursively applied till we reach node x from r; at this point, we have eliminated every node in T that is not in S, as a candidate for the successor of x.

(b) Node x is in the subtree rooted at the right child of r - Observe that the $key[r]$ and the keys of all the nodes in its left subtree are smaller than the keys of any node in $S \cup \{x\}$ and therefore cannot be candidates for the successor of x. This property continues along the path P, till such time as an ancestor of x, viz. z, is reached, such that x is in the left subtree of z. Now, by the property of binary search trees, $key[z]$ and the keys of all the nodes in its right subtree are greater than the key of any node in $S \cup \{x\}$; thus they can also be excluded from the candidate list of possible successors of x. The above two observations are recursively applied till we reach node x from r; at this point, we have eliminated every node in T that is not in S, as a candidate for the successor of x.

□

We have thus established that the succeesor of x, must lie in the subtree rooted at y. Let z denote the successor of x and let it have a left subchild, say p. By the property of binary search trees, $key[x] < key[p] < key[z]$, contradicting the choice of z, as the successor of x. □

5. `Dynamic Programming:` Let $S = \{s_1, s_2, \ldots, s_n\}$ denote a collection of n positive numbers, such that $\sum_{i=1}^{n} s_i = N$. Devise an algorithm that runs in time $O(n \cdot N)$ to check if there is a set $S' \subseteq S$, such that $\sum_{s_i \in S'} s_i = \sum_{s_i \in S - S'} s_i$. (*Hint: 0/1 Knapsack!*)

Solution: Observe that if $\sum_{i=1}^{n} s_i = N$ is an odd number the answer is immediately "no", since an odd number cannot be broken into two equal integral parts.

Associate a decision variable x_i with each s_i, where

$$\begin{aligned} x_i &= 1, \;\; if \;\; s_i \in S' \\ &= 0, \;\; if \;\; s_i \notin S' \end{aligned}$$

Thus, a sequence of decisions have to be made on variables x_1 through x_n.

We define $m[i, j]$ to be **T** (true), if some subset of the elements in $\{s_1, s_2 \ldots, s_i\}$ has elements that add up to j. In this notation, $m[n, \frac{M}{2}]$ is the answer to our question, i.e., the answer to the input problem is "yes" if and only if $m[n, \frac{M}{2}]$ is **T**.

The key observation is that $m[i, j]$ can be true if and only if one of the following holds:

- $m[i-1, j]$ is **T**. Clearly if there is a subset of the first $i-1$ elements that sums to j, the same subset can be used as the subset of the first i elements that sums to j. This corresponds to the case of assigning $x_i = 0$;
- $m[i-1, j-s_i]$ is **T**. If s_i is to be included in the subset of $\{s_1, s_2, \ldots, s_i\}$ that sums to j, then there must exist some subset of the first $i-1$ elements that sums to $j - s_i$. This corresponds to the case of assigning $x_i = 1$.

Proceeding this way, we can build a table $m[1..n, 0..\frac{N}{2}]$ and check whether $m[n, \frac{M}{2}]$ is **T**. Since each entry can be filled in $O(1)$ time, the total time taken is $O(n \cdot N)$.

It is very important to note that the recursive algorithm developed above does not run in time that is polynomial in the size of the input. The input size is $\log s_1 + \log s_2 + \ldots + \log s_n$, whereas the running time is $O(n \cdot \sum_{i=1}^{n} s_i)$.

□

A An Alternative Solution to Problem 1

Lower Bound:
Assume, without loss of generality, that n is even. Observe that we can obtain a lower bound on this series as follows.

$$\begin{aligned}
\sum_{k=1}^{n} k^2 \log k &= 1^2 \cdot \log 1 + 2^2 \cdot \log 2 + \ldots + n^2 \cdot \log n \\
&\geq \sum_{k=\frac{n}{2}}^{n} k^2 \log k \\
&= (\frac{n}{2})^2 \cdot \log(\frac{n}{2}) + (\frac{n+1}{2})^2 \cdot \log(\frac{n+1}{2}) + \ldots + n^2 \cdot \log n \\
&\geq (\frac{n}{2})^2 \cdot \log(\frac{n}{2}) + (\frac{n}{2})^2 \cdot \log(\frac{n}{2}) + \ldots + (\frac{n}{2})^2 \cdot \log(\frac{n}{2}) \\
&= \frac{n}{2} \cdot (\frac{n}{2})^2 \cdot \log(\frac{n}{2}) \\
&= \frac{n}{2} \cdot \frac{n^2}{4} \cdot (\log n - \log 2) \\
&= \frac{n^3}{8} \cdot (\log n - 1) \\
&= \frac{n^3}{8} \cdot \log n - \frac{n^3}{8} \\
&\geq \frac{1}{10} n^3 \log n \qquad (\text{for } n \geq 32)
\end{aligned}$$

Note:

$$\begin{aligned}
\frac{n^3}{8} \cdot \log n - \frac{n^3}{8} &\geq \frac{1}{10} \cdot n^3 \cdot \log n \\
10 \cdot \frac{n^3}{8} \cdot \log n - 10 \cdot \frac{n^3}{8} &\geq n^3 \cdot \log n \\
\frac{5}{4} \cdot n^3 \log n - \frac{5}{4} \cdot n^3 &\geq n^3 \cdot \log n \\
\frac{5}{4} \cdot n^3 \log n - n^3 \log n &\geq \frac{5}{4} \cdot n^3 \\
\frac{1}{4} \cdot n^3 \log n &\geq \frac{5}{4} \cdot n^3 \\
\log n &\geq 5 \\
n &\geq 2^5 \\
n &\geq 32
\end{aligned}$$

We can choose $n \geq 32$. Thus, $\sum_{k=1}^{n} k^2 \cdot \log k \in \Omega(n^3 \cdot \log n)$.
Upper Bound:
Observe that we can obtain an upper bound on this series by bounding each term of the series, by the largest term ($n^2 \cdot \log n$). From this, we have:

$$\begin{aligned}
\sum_{k=1}^{n} k^2 \cdot \log k &= 1^2 \cdot \log 1 + 2^2 \cdot \log 2 + \ldots + n^2 \cdot \log n \\
&\leq n^2 \cdot \log n + n^2 \cdot \log n + \ldots + n^2 \cdot \log n
\end{aligned}$$

$$
\begin{aligned}
&= n \cdot n^2 \log n \\
&= n^3 \cdot \log n
\end{aligned}
$$

Thus, $\sum_{k=1}^{n} k^2 \cdot \log k \in O(n^3 \log n)$.

Observe that $\sum_{k=1}^{n} k^2 \cdot \log k$ is bounded below by $\Omega(n^3 \cdot \log n)$ and it is bounded above by $O(n^3 \cdot \log n)$. Therefore, $\sum_{k=1}^{n} k^2 \cdot \log k \in \Theta(n^3 \log n)$.