Design of Algorithms - Homework I (Solutions)

K. Subramani
LCSEE,
West Virginia University,
Morgantown, WV
{ksmani@csee.wvu.edu}

1 Problems

1. Consider the following algorithm for sorting an array of n numbers.

Function ARRAY-SORT($\mathbf{A}$, n)

```
for (i = 1 to n) do
  for (j = n downto i + 1) do
    if (A[j] < A[j − 1]) then
      SWAP(A[j], A[j − 1]).
    end if
  end for
end for
```

Algorithm 1.1: Sorting Algorithm

Argue the correctness of the algorithm using loop invariants and analyze its running time.

Solution:

Correctness: The first step is to choose a useful loop invariant.

Consider the following invariant: *At the end of the* k^{th} *iteration of the outer* **for** *loop, the array* $A[1 \cdot \cdot k]$ *is sorted and* $A[k]$ *is the* k^{th} *smallest element in* $\mathbf{A}$.

Let us focus on the first execution of the outer **for** loop. In this case, the inner **for** loop executes $(n-1)$ times. After the inner **for** loop has executed for the first time, $A[n-1]$ stores the minimum of $A[n]$ and $A[n-1]$. After it has executed the second time, $A[n-2]$ stores the minimum of $A[n], A[n-1]$ and $A[n-2]$. Likewise, when the inner **for** loop has executed for the $(n-1)^{th}$ time, $A[1]$ stores the minimum of $A[n], A[n-1], \ldots, A[1]$. It thus follows that at the end of the first iteration, $A[1]$ is the smallest element in $\mathbf{A}$.

Assume that the loop invariant holds at the end of the r^{th} iteration, i.e., assume that after the outer **for** loop has executed r times, the array $A[1 \cdot \cdot r]$ is sorted and $A[r]$ is the r^{th} smallest element of the array. We shall now argue that the loop invariant holds after the $(r+1)^{th}$ iteration. Observe that as argued before, the inner **for** loop moves the minimum element of the array $A[(r+1) \cdot \cdot n]$ into $A[r+1]$ performing swaps as necessary. It follows that $A[r+1]$ is now the $(r+1)^{th}$ smallest element of the array and that the array $A[1 \cdot \cdot (r+1)]$ is sorted.

The algorithm terminates when the outer **for** loop has executed n times. At this juncture, we can conclude that the array $A[1 \cdot \cdot n]$ is sorted and that $A[n]$ is the n^{th} smallest element in the array.

Resource Analysis: Let $T(n)$ denote the running time of the algorithm on an array of n elements. Observe that we are interested in the number of element to element comparisons only.

Since there is only one element to element comparison, in the nested **for** loops, we have,

$$
\begin{aligned}
T(n) &= \sum_{i=1}^{n} \sum_{j=i+1}^{n} 1 \\
&= \sum_{i=1}^{n} (n - (i+1) + 1) \\
&= \sum_{i=1}^{n} (n - i) \\
&= \sum_{i=1}^{n} n - \sum_{i=1}^{n} i \\
&= n \cdot n - \frac{n \cdot (n+1)}{2} \\
&= \frac{n \cdot (n-1)}{2}
\end{aligned}
$$

□

2. (a) Show that $\log n! = \Theta(n \cdot \log n)$.
   (b) Show that $\max(f(n), g(n)) = \Theta(f(n) + g(n))$, where $f(n)$ and $g(n)$ are non-negative functions.

Solution:

(a) We first show that $\log n! = O(n \cdot \log n)$. Observe that,

$$
\begin{aligned}
\log n! &= \log(1 \cdot 2 \cdot \ldots n) \\
&= \log 1 + \log 2 + \ldots \log n \\
&\leq \log n + \log n + \ldots \log n \\
&= n \cdot \log n
\end{aligned}
$$

We next show that $\log n! = \Omega(n \cdot \log n)$. Observe that,

$$
\begin{aligned}
\log n! &= \log(1 \cdot 2 \cdot \ldots n) \\
&= \log 1 + \log 2 + \ldots \log n \\
&\geq \log(\frac{n}{2} + 1) + \log(\frac{n}{2} + 2) + \ldots \log(\frac{n}{2} + \frac{n}{2}) \\
&\geq \log \frac{n}{2} + \log \frac{n}{2} + \ldots \log \frac{n}{2} \\
&= \frac{n}{2} \log \frac{n}{2} \\
&= \frac{n}{2} \log n - \frac{n}{2}
\end{aligned}
$$

Now note that,

$$
\frac{n}{2} \log n - \frac{n}{2} \quad \geq \quad \frac{n}{4} \log n
$$

as long as,

$$
\frac{n}{2} \log n - \frac{n}{4} \log n \quad \geq \quad \frac{n}{2}
$$

$$
\begin{aligned}
\Rightarrow \frac{n}{4}\log n &\geq \frac{n}{2}\\
\Rightarrow \log n &\geq 2\\
\Rightarrow n &\geq 4
\end{aligned}
$$

Thus, $\log n! = \Omega(n \cdot \log n)$.

From the above discussion, we can conclude that, $\log n! = \Theta(n \cdot \log n)$.

(b) First observe that, $\max(f(n), g(n)) \leq (f(n) + g(n))$, as long as $f(n)$ and $g(n)$ are non-negative functions, i.e., $\max(f(n), g(n)) = O(f(n) + g(n))$.

Now observe that, $(f(n) + g(n)) \leq 2 \cdot \max(f(n), g(n))$, as long as $f(n)$ and $g(n)$ are non-negative functions, i.e, $\max(f(n), g(n)) = \Omega((f(n) + g(n))$.

From the above discussion, we can conclude that, $\max(f(n) + g(n)) = \Theta(f(n) + g(n))$.

□

3. An $m \times n$ array $\mathbf{A}$ of integers is said to be a **Monge Array**, if for all i, j, k, and l, such that $1 \leq i < k \leq m$ and $1 \leq j < l \leq n$, we have,
$$A[i,j] + A[k,l] \leq A[i,l] + A[k,j]$$

(a) Prove that an array $\mathbf{A}$ is Monge, if and only if for all $i = 1, 2, \ldots, m-1$ and $j = 1, 2, \ldots n-1$, we have,
$$A[i,j] + A[i+1, j+1] \leq A[i, j+1] + A[i+1, j]. \tag{1}$$

Solution: If we assume that $\mathbf{A}$ is Monge, then it is easy to see that Condition (1) holds, that is:
$$A[i,j] + A[i+1, j+1] \leq A[i, j+1] + A[i+1, j]$$
(just let $k = i+1$ and $l = j+1$).

For the other direction, we first prove a "helper" lemma.

Lemma 1.1 *Let $n \geq 2$ and let $\mathbf{A}$ be a $2 \times n$ matrix satisfying Condition (1). Then $\mathbf{A}$ is Monge.*

Proof: By induction on n. The base case is $n = 2$, so $\mathbf{A}$ is a 2×2 matrix. In this case, Condition (1) and the Monge definition coincide.

For the inductive step, assume that the lemma holds for all $c \leq n$ and let $\mathbf{A}$ be a $2 \times (n+1)$ matrix which satisfies Condition (1) (this is *strong induction*). We need to show that
$$A[1,j] + A[2,l] \leq A[1,l] + A[2,j] \tag{2}$$
for all $1 \leq j < l \leq (n+1)$. If $j \neq 1$ or $l \neq (n+1)$, we can apply the induction hypothesis to a sub-matrix of $\mathbf{A}$.

If $j = 1$ and $l = (n+1)$, consider the following entries of $\mathbf{A}$:
$$\begin{bmatrix} A[1,1] & \cdots & A[1, l-1] & A[1,l] \\ A[2,1] & \cdots & A[2, l-1] & A[2,l] \end{bmatrix}.$$

The sub-matrix of $\mathbf{A}$ consisting of the first n columns satisfies Condition (1) and is therefore Monge, by the induction hypothesis. Similary, the sub-matrix consisting of the last two columns of $\mathbf{A}$ form a Monge sub-matrix. Since both of these submatrices are Monge, the following inequalities hold:
$$A[1,1] + A[2, l-1] \leq A[1, l-1] + A[2,1],$$
$$A[1, l-1] + A[2,l] \leq A[1,l] + A[2, l-1].$$

Adding these two inequalites together yields Condition (2). □

We now prove that all matrices which satisfy Condition (1) are, in fact, Monge. Consider the following ordering on $\mathbb{N} \times \mathbb{N}$:

$$(n, m) < (n', m') \text{ precisely when } [(n < n') \text{ or } (n = n' \text{ and } m < m')].$$

(that is, the "dictionary order" on $\mathbb{N} \times \mathbb{N}$). It is easy to see that this is a well-ordering on $\mathbb{N} \times \mathbb{N}$. Let (n, m) be the least element in this ordering such that there exists an $n \times m$ matrix $\mathbf{A}$ which satisfies Condition (1) but is not Monge. We make the following observations about the 2-tuple (n, m):

(i) Both n and m must be at least 2. If m or n is 1, the Monge condition is vacously true.

(ii) n must be at least 3, since if n is 2, $\mathbf{A}$ would be Monge by Lemma 1.1.

Therefore, we can assume that $n \geq 3$ and $m \geq 2$.

The Monge condition requires the entries of $\mathbf{A}$ to satisfy many inequalities. Note that the minimality of (n, m) guarantees that all of these inequalities, except for

$$A[1,1] + A[m,n] \leq A[1,m] + A[n,1] \tag{3}$$

hold. Also note that the minimality of (n, m) does not immediately imply anything about Condition (3). Intuitively, we are guaranteed that all of the inequalities, except for possibly the one involving the "corners" of $\mathbf{A}$, are satisfied. If this is not the case, we could find a smaller submatrix within $\mathbf{A}$ that satisfies Condition (1) but is not Monge. This would contradict the minimality of (n, m).

We now show that Condition (3) must also hold. Consider the following entries of $\mathbf{A}$:

$$\begin{bmatrix} A[1,1] & \cdots & A[1,n] \\ \vdots & \vdots & \vdots \\ A[n-1,1] & \cdots & A[n-1,m] \\ A[n,1] & \cdots & A[n,m] \end{bmatrix}$$

We know that the inequalities

$$A[1,1] + A[n-1,m] \leq A[1,n] + A[n-1,1],$$

$$A[n-1,1] + A[n,m] \leq A[n-1,m] + A[n,1],$$

must hold. (Why?) Adding these inequalities together yields

$$A[1,1] + A[m,n] \leq A[1,m] + A[n,1].$$

Thus $\mathbf{A}$ is in fact a Monge array, which contradicts our hypothesis. □

(b) Let $f(i)$ be the index of the column containing the leftmost minimum element of row i. Prove that $f(1) \leq f(2) \leq \ldots \leq f(m)$, for any $m \times n$ Monge array.

Solution: Let $\mathbf{A}$ denote a Monge array, such that that there exists an i with $f(i) > f(i+1)$. We can then draw the following picture of $\mathbf{A}$:

$$\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & A[i, f(i+1)] & \cdots & A[i, f(i)] & \cdots \\ \cdots & A[i+1, f(i+1) & \cdots & A[i+1, f(i)] & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

Observe that, we must have, $A[i, f(i+1)] > A[i, f(i)]$, since if $A[i, f(i+1)] = A[i, f(i)]$, then $f(i)$ is not the index of the column containing the leftmost minimum element of row i.

Likewise, $A[i+1, f(i)] \geq A[i+1, f(i+1)]$, since $f(i+1)$ is the index of the column containing the leftmost minimum element of row $i+1$.

Adding these inequalities together, we get,

$$A[i, f(i+1)] + A[i+1, f(i)] > A[i, f(i)] + A[i+1, f(i+1)]$$

This contradicts the fact that $\mathbf{A}$ is Monge. □

4. Show that for any integer $n \geq 0$,

$$\sum_{k=0}^{n} C(n,k) \cdot k = n \cdot 2^{n-1}.$$

Solution: At $n = 0$, the LHS of the identity is $C(n,0) \cdot 0 = 0$ and the RHS is $0 \cdot 2^{-1} = 0$.
At $n = 1$, the LHS of the identity is $C(n,0) \cdot 0 + C(n,1) \cdot 1 = n$ and the RHS is $n \cdot 2^{1-1} = n$.
Thus, the identity is clearly true for $n = 0$ and $n = 1$. For $n \geq 2$, we consider two cases:

(i) n is odd - In this case, the numbers $\frac{n-1}{2}$, $\frac{n+1}{2}$, etc. are integral.
Observe that,

$$\begin{aligned}
\sum_{k=0}^{n} C(n,k) \cdot k &= 0 \cdot C(n,0) + 1 \cdot C(n,1) + 2 \cdot C(n,2) + \cdots + (n-1) \cdot C(n,n-1) + n \cdot C(n,n) \\
&= [0 \cdot C(n,0) + n \cdot C(n,n)] + [1 \cdot C(n,1) + (n-1) \cdot C(n,n-1)] \\
&\quad +[2 \cdot C(n,2) + (n-2) \cdot C(n,n-2)] + \ldots \\
&\quad [(\frac{n-1}{2}) \cdot C(n, \frac{n-1}{2}) + (\frac{n-1}{2}+1) \cdot C(n, (\frac{n-1}{2}+1)] \\
&= \sum_{k=0}^{\frac{n-1}{2}} [k \cdot C(n,k) + (n-k) \cdot C(n,n-k)] \\
&= \sum_{k=0}^{\frac{n-1}{2}} [k \cdot C(n,k) + (n-k) \cdot C(n,k)], \text{ since } C(n,k) = C(n,n-k) \\
&= \sum_{k=0}^{\frac{n-1}{2}} [n \cdot C(n,k)] \\
&= n \cdot \sum_{k=0}^{\frac{n-1}{2}} C(n,k)
\end{aligned}$$

Let us focus on the sum $\sum_{k=0}^{\frac{n-1}{2}} C(n,k)$. In class, we showed that, $\sum_{k=0}^{n} C(n,k) = 2^n$. Now note that,

$$\begin{aligned}
\sum_{k=0}^{\frac{n-1}{2}} C(n,k) &= C(n,0) + C(n,1) + \ldots C(n, \frac{n-1}{2}) \\
&= \frac{1}{2} \cdot [2 \cdot C(n,0) + 2 \cdot C(n,1) + \ldots 2 \cdot C(n, \frac{n-1}{2})] \\
&= \frac{1}{2} \cdot [(C(n,0) + C(n,n)) + (C(n,1) + C(n,n-1) + \ldots (C(n, \frac{n-1}{2}) + C(n, \frac{n+1}{2}))], \\
&\quad \text{since } C(n,k) = C(n,n-k) \\
&= \frac{1}{2} \sum_{k=0}^{n} C(n,k) \\
&= \frac{1}{2} 2^n
\end{aligned}$$

Accordingly,

$$\sum_{k=0}^{n} C(n,k) \cdot k \quad = \quad n \cdot \sum_{k=0}^{\frac{n-1}{2}} C(n,k)$$

$$= \quad n \cdot (\frac{2^n}{2})$$
$$= \quad n \cdot 2^{n-1}$$

(ii) n is even - In this case, the number $\frac{n}{2}$ is integral. The analysis is identical to the case when n is odd, except that the term $\frac{n}{2}C(n, \frac{n}{2})$ is not combined with any other term. Accordingly, we have,

$$\begin{aligned}
\sum_{k=0}^{n} C(n,k) \cdot k &= n \cdot \sum_{k=0}^{\frac{n}{2}} C(n,k) \\
&= n \cdot \frac{2^n}{2} \\
&= n \cdot 2^{n-1}
\end{aligned}$$

□

5. Let X be a non-negative random variable and suppose that $E[X]$ and $\sigma = \sqrt{Var(X)}$ are well-defined.
   (a) Show that $Pr[X \geq t] \leq \frac{E[X]}{t}$, for all $t > 0$.
   (b) Show that $Pr[|X - E[X]| \geq t \cdot \sigma] \leq \frac{1}{t^2}$, for any $t > 0$.

Solution:

(a) Observer that,

$$\begin{aligned}
E[X] &= \sum_{x} x \cdot Pr[X = x] \\
&= \sum_{0 \leq x < t} x \cdot Pr[X = x] + \sum_{x \geq t} x \cdot Pr[X = x] \\
&\geq \sum_{x \geq t} x \cdot Pr[X = x], \text{ since } X \geq 0 \\
&\geq \sum_{x \geq t} t \cdot Pr[X = x] \\
&= t \cdot Pr[X \geq t] \\
\Rightarrow Pr[X \geq t] &\leq \frac{E[X]}{t}
\end{aligned}$$

The above inequality is known as Markov's inequality.

(b) Observe that,

$$\begin{aligned}
Pr[|X - \mathbf{E}[X]| \geq t \cdot \sigma] &= Pr[|X - E[X]|^2 \geq t^2 \cdot \sigma^2] \\
&\leq \frac{E[(X - E[X])^2]}{t^2 \cdot \sigma^2}], \text{ by Markov's inequality} \\
&= \frac{Var[X]}{t^2 \cdot \sigma^2}, \text{ by the definition of variance} \\
&= \frac{\sigma^2}{t^2 \cdot \sigma^2} \\
\Rightarrow Pr[|X - E[X]| \geq t \cdot \sigma] &\leq \frac{1}{t^2}
\end{aligned}$$

□