CECS 528, Exam 1, yellow Version, Fall 2023, Dr. Ebert

Rules for Completing the Problems

NO NOTES, BOOKS, ELECTRONIC DEVICES, OR INTERPERSONAL COMMUNICATION allowed when solving these problems. Make sure all these items are put away BEFORE looking at the problems. FAILURE TO ABIDE BY THESE RULES MAY RESULT IN A FINAL COURSE GRADE OF F.

Directions

Choose up to **six problems** to solve. Clearly mark each problem you want graded by placing an X or check mark in the appropriate box in the Grade(?) row of the table below. **If you don't mark any problems for us to grade or mark 7 or more problems, then we will record grades for the 6 that received the *fewest* points.**

Problem	1	2	3	4	5	6
Grade?						
Result						

Your Full Name:

Your Class ID:

1. Solve each of the following problems. Note: correctly solving these problems counts for passing LO1.

   a. Show each of the subproblem instances that must be solved when using the recursive multiplication algorithm for finding the product xy. Do this for $x = 19$ and $y = 35$. Make sure to provide the solution to each subproblem instance. Hint: there are *seven* subproblem instances, including the original problem instance. (10 pts)

   Solution.

x	y	xy
19	35	$646 + 19 = 665$
19	17	$304 + 19 = 323$
19	8	152
19	4	76
19	2	38
19	1	19
19	0	0

   b. Consider the RSA key set $(N = 143 = 11 \cdot 13, e = 7)$. Determine the decryption key d. (15 pts)

   Solution. We want the multiplicative inverse of 7 mod $10 \cdot 12 = 120$. We have

$$120 = (7)(17) + 1 \Rightarrow 120 + 7(-17) = 1 \Rightarrow$$

$$7(-17) \equiv 1 \bmod 120.$$

   Therefore, -17 is the multiplicative inverse of 7, and which is congruent to 103 mod 120.

2. Solve each of the following problems. Note: correctly solving these problems counts for passing LO2.

   a. Use the Master Theorem to determine the growth of $T(n)$ if it satisfies the recurrence $T(n) = 4T(n/2) + n^{\log_3 9} \log^2 n$. (10 pts)

   Solution. By Case 4 of the Master Theorem, $T(n) = \Theta(n^2 \log^3 n)$.

   b. Use the substitution method to prove that, if $T(n)$ satisfies

$$T(n) = 3T(n/2) + n \log n,$$

   Then $T(n) = \mathrm{O}(n^2)$. (15 pts)

   Solution. Inductive assumption: $T(k) \leq Ck^2$ for all $k < n$ and some constant $C > 0$. Then

$$T(n) = 3T(n/2) + n \log n \leq 3C(n/2)^2 + n \log n = \frac{3}{4}Cn^2 + n \log n \leq Cn^2 \Leftrightarrow$$

$$\frac{C}{4}n^2 \geq n \log n \Leftrightarrow C \geq \frac{4 \log n}{n},$$

   which is true for any $C \geq 1$, so long as $n \geq 16$. In general, C can be any positive constant so long as n is sufficiently large.

3. Solve each of the following problems. Note: correctly solving these problems counts for passing LO3.

   a. Recall that the `Find-Statistic` algorithm makes use of the Partitioning algorithm and uses a pivot that is guaranteed to have at least

$$3(\lfloor\frac{1}{2}\lceil\frac{n}{5}\rceil\rfloor - 2) \geq 3(\frac{1}{2}\cdot\frac{n}{5} - 3) = \frac{3n}{10} - 9.$$

   members of array a both to its left and to its right. Rewrite the above inequalities but now assuming the algorithm uses groups of nine instead of groups of five. Explain your reasoning for each of the numerical changes that you make.

   Solution. Replace the 5 with 9 since now the groups have size nine. Also, replace the 3 with 5 since every group has at least five members that are greater than or equal to (respectively, less than or equal to) the chosen pivot (i.e. median of medians). Thus we have

$$5(\lfloor\frac{1}{2}\lceil\frac{n}{9}\rceil\rfloor - 2) \geq 5(\frac{1}{2}\cdot\frac{n}{9} - 3) = \frac{5n}{18} - 15.$$

   b. Use Strassen's products $P_1 = a(f-h) = af - ah$, $P_2 = (a+b)h = ah + bh$, $P_3 = (c+d)e = ce + de$, $P_4 = d(g-e) = dg - de$, $P_5 = (a+d)(e+h) = ae + ah + de + dh$, $P_6 = (b-d)(g+h) = bg + bh - dg - dh$, $P_7 = (a-c)(e+f) = ae - ce - cf + af$. to compute the matrix product

$$\begin{pmatrix} 2 & -3 \\ -5 & 4 \end{pmatrix}\begin{pmatrix} 3 & 1 \\ -4 & 5 \end{pmatrix}$$

   Show all work. (13 pts)

Solution. We have

$$ae + bg = P_5 + P_6 - P_2 + P_4 = 48 + (-7) - (-5) + (-28) = 18,$$

$$af + bh = P_1 + P_2 = -8 + (-5) = -13,$$

$$ce + dg = P_3 + P_4 = -3 + (-28) = -31,$$

and

$$cf + dh = -P_7 + P_5 + P_1 - P_3 = -28 + 48 + (-8) - (-3) = 15,$$

to yield the product matrix

$$\begin{pmatrix} 18 & -13 \\ -31 & 15 \end{pmatrix}.$$

4. Recall the following recursive multiplication algorithm for multiplying m-bit integer x with n-bit integer y.

```
Name: multiply
Inputs: $m$-bit integer $x$ and $n$-bit integer $y$.
Output: $xy$.
Begin
If $y = 0$, then return 0.
$z = \texttt{multiply}(x, \lfloor y/2 \rfloor)$.
If $y$ is even, then return $2z$.
Else return $x + 2z$.
End
```

Determine the worst case running time of this algorithm assuming that

a. division by 2 requires $\mathrm{O}(1)$ steps,

b. multiplying a k-bit integer by 2 requires $\mathrm{O}(k)$ steps and produces a $(k+1)$-bit integer,

c. adding x to a k-bit integer requires $\mathrm{O}(m+k)$ steps and produces a $(\max(k,m)+1)$-bit integer.

Show all work. (25 pts)

Solution. The worst case occurs when y is odd in each subproblem, since we must add x in that case. Since (starting with $y = 0$ and $i = 0$) the i th subproblem has a multiplier with i bits, the i th product will have at most $m+i$ bits and computing this product requires at total of $\mathrm{O}(2(m+(i-1))) = \mathrm{O}(m+i)$ steps. Moreover,

$$\sum_{i=0}^{n}(m+i) = \mathrm{O}(mn+n^2)$$

steps.

5. Let a, b, m all be positive integers and suppose $(a, m) \mid b$. Prove that the equation

$$ax \equiv b \bmod m$$

has a solution. (25 pts)

Solution. By Proposition 3.3, we know that, since $(a, m) \mid b$, there are integers x and y for which

$$ax + my = b,$$

which implies

$$ax \equiv b \bmod m,$$

and so any member of the residue class $(x)_m$ is a solution to the equation.

6. Consider a problem that takes as input an array a of $n \geq 2$ integers, and a positive integer m that divides evenly into n, i.e. $n = md$ for some positive integer d. The problem is to find the $k = m, 2m, \ldots, (d-1)m$ statistics of a. Provide a high-level description of an algorithm that solves this problem in $\mathrm{O}(n \log d)$ steps. Prove that your algorithm achieves the desired running time. Note: credit will not be awarded to descriptions that are ambiguous, make incorrect assumptions/conclusions, and/or do not achieve the desired running time. (25 pts)

   Solution. We modify the `Find-Statistic` algorithm in three ways: i) allow for an array of k-values rather than just one k-value, ii) always use the median of a as pivot when performing the `Partitioning` algorithm, and iii) return an array of statistics, rather than a single statistic. Moreover, since the k-values are multiples of m and spread throughout the array, the algorithm will require making two recursive calls: one on a_{left} and one on a_{right}. This will be true until the array size is small enough to where it has at most one multiple of m. This occurs at depth r, where

$$n/2^r \leq m$$

   which implies that

$$r \geq \log(m/n) = \log d,$$

   where $md = n$. Moreover, when the subproblem has a single k-value, then we may revert back to the original `Find-Statistic` algorithm and solve it using $\mathrm{O}(m)$ steps since the array size is at most m. Thus the algorithm's running time is

$$\mathrm{O}(n \log d) + (d-1)\mathrm{O}(m) = \mathrm{O}(n \log d) + \mathrm{O}(n),$$

   where $\mathrm{O}(n \log d)$ is from the fact that $\mathrm{O}(n)$ total amount of work is performed in each of the first d levels of the recursion tree. In other words, the recursion tree is a perfect binary tree down to depth $r = \lceil \log d \rceil$.