CS 6363: Design and Analysis of Algorithms
Exam #3, December 5, 2018
Professor D.T. Huynh

Student Name:	KEY

General Remarks. This exam comprises 4 problems:

Problem ♯1 is assigned 25 points,

Problem ♯2 is assigned 25 points,

Problem ♯3 is assigned 25 points, and

Problem ♯4 is assigned 25 points.

Thus, the maximum score is 100 points.

Unless explicitly stated, *no correctness proofs* are required for your algorithms and (time) complexity means worst-case complexity.

Provide clean answers on the exam booklet. Use aditional paper only when necessary.

This is a **closed-book** exam

Exam time: 10:00am – 11:20 am

Good Luck!

#1	#2	#3	#4	Total

Problem # 1. (Closest pair of points)

1. Describe the algorithm to compute for n given points in the plane $\{p_1 = (x_1, y_1), \ldots, p_n = (x_n, y_n)\} = P$ the closest pair. Here the distance between $p_i = (x_i, y_i)$ and $p_j = (x_j, y_j)$ is defined as $Max\{|x_i - x_j|, |y_i - y_j|\}$. Argue that your algorithm works correctly.
2. Give a detailed analysis of the running time of the algorithm!

The algorithm is the same as described in class. The only modification is: the distance is computed as

$$d(p_i, p_j) = Max\{|x_i - x_j|, |y_i - y_j|\}$$

1. (a) Let array X be P sorted by x-coordinates
Y be ——— y-coordinates
If $|P| \le 3$, find closest pair directly

(b) Divide Step: $|P| = n$.

15

(i) Find a vertical line L using median of X to divide P into P_L, P_R s.t. $|P_L| = \lceil \frac{n}{2} \rceil$, $|P_R| = \lfloor \frac{n}{2} \rfloor$

(ii) Let X_L (X_R) and Y_L (Y_R) be arrays associated with P_L (P_R) obtained by unmerging X, Y

(c) Recursive Step:
Compute closed pairs of P_L using $X_L, Y_L \to \delta_L$
and P_R ——— $X_R, Y_R \to \delta_R$

(d) Conquer Step: Let $\delta = Min\{\delta_L, \delta_R\}$
Let P′ be set of points within 2δ-wide strip centered at line L with Y′ obtained from Y by unmerging
For each $p \in P'$ in increasing order of y-coord. in Y′ search for closest pair among p and the next 7 neighbors in P′; update closest pair

Correctness: Similar to lecture notes!

2.

(a) : $O(n \log n)$

$$\left.\begin{array}{ll} \text{(b)} : & O(n) \quad (\text{unmerging is } O(n)) \\ \text{(c)} : & 2T(\frac{n}{2}) \\ \text{(d)} : & O(n) \end{array}\right\} T(n)$$

$\Rightarrow$ Total running time for (b) – (d):

10

$$T(n) = 2T(\tfrac{n}{2}) + O(n)$$
$$= O(n \log n)$$

$\Rightarrow$ Overall running time:

$$\underbrace{O(n \log n)}_{(a)} + \underbrace{T(n)}_{(b)-(d)} = O(n \log n)$$

Problem ♯ 2.

6

1. The DFT of a vector $\mathbf{a} = (a_0, a_1, a_2, \ldots, a_{n-1})$ is defined as the product $A \times \mathbf{a}$, where A is an $n \times n$ matrix. Let ω be the complex principal $n-$th root of unity. Define the matrix A and its inverse A^{-1}.

$$A = (a_{ij}) = (\omega^{ij}) \qquad A^{-1} = \frac{1}{n}(\omega^{-ij})$$
$$0 \leqslant i,j \leqslant n-1 \qquad 0 \leqslant i,j \leqslant n-1$$

6

2. What is polynomial evaluation, interpolation and their relation to DFT and DFT^{-1}?

$p(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1}$, $a = (a_0, \ldots, a_{n-1})$

evaluation: evaluating $p(x)$ at n points

$= DFT(a) = A.a = y$ if points $= \omega^0, \ldots, \omega^{n-1}$

interpolation: recovering a from values of p at n points

if $y = (y_0, \ldots, y_{n-1})$ are values of p at $\omega^0, \ldots, \omega^{n-1}$

then $a = DFT^{-1}(y)$

$= A^{-1}.y$

6

3. Let $p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}$ be a polynomial of degree $n-1$. Describe the technique of evaluating p at a point $x = c$ recursively based on two polynomials $p_0(y)$ and $p_1(y)$ of degree $n/2 - 1$ obtained from $p(x)$.

Define $p_0(y) = a_0 + a_2 y + \cdots + a_{n-2} y^{\frac{n}{2}-1}$

$p_1(y) = a_1 + a_3 y + \cdots + a_{n-1} y^{\frac{n}{2}-1}$

Then $p(x) = p_0(x^2) + x\, p_1(x^2)$

7

4. Using the technique described above evaluate the polynomial $1 + 5x - 7x^2 + 2x^3 - 5x^4 + 4x^5$ at $x = 2$. (Show your steps!) (You may evaluate the subpolynomials $p_0(y)$ and $p_1(y)$ of degree 2 directly)

$p_0(y) = 1 - 7y - 5y^2$; $p_1(y) = 5 + 2y + 4y^2$

$x = 2 \Rightarrow x^2 = 4$

$$\Rightarrow p(x) = p_0(4) + 2 p_1(4)$$
$$= (1 - 7 \times 4 - 5 \times 4^2) + 2(5 + 2 \times 4 + 4 \times 4^2)$$
$$= -107 + 2 \times 77$$
$$= 47.$$

Problem ♯ 3.

1. Define the classes **P**, **NP**, polynomial-time reducibility, and **NP**-complete language.

P = class of problems solvable by poly-time DTMs

NP = ______ NTMs

$L_1 \le_p L_2$ iff $\exists$ poly-time computable function $f: \Sigma^* \to \Sigma^*$ s.t. $w \in L_1 \Leftrightarrow f(w) \in L_2$

8

L is NP-hard if $\forall L' \in NP: L' \le_p L$

L is NP complete if (1) $L \in NP$

(2) L is NP-hard

2. Describe the procedure that converts a given CNF Boolean formula into an equivalent 3CNF.

Let F be in CNF. Consider a clause $C = \ell_1 \vee \dots \vee \ell_m$, $m > 3$ containing more than 3 literals $\ell_1, \dots, \ell_m$. Let $y_1, \dots, y_m$ be new variables and define

9

$$C' = (\ell_1 \vee \overline{y_1}) \wedge (y_1 \vee \ell_2 \vee \overline{y_2}) \wedge \dots \wedge (y_{m-1} \vee \ell_m \vee \overline{y_m}) \wedge y_m$$

Claim, Let $\psi : \{x_1, \dots, x_n\} \to \{0,1\}$ be a truth assigment. Then $\exists$ an extension $\varphi : \{x_1, \dots, x_n, y_1, \dots, y_m\} \to \{0,1\}$ s.t. $\psi(C) = 1 \Leftrightarrow \varphi(C') = 1$

3. Using the procedure in (2) transform the following CNF formula into 3CNF:

$$(x_1 \vee \neg x_3 \vee \neg x_4 \vee x_5 \vee \neg x_6) \wedge \underbrace{(x_3 \vee \neg x_4 \vee x_6) \wedge (\neg x_1 \vee x_4)}_{\text{in 3-CNF}}$$

We only need to convert

$$C = (x_1 \vee \overline{x_3} \vee \overline{x_4} \vee x_5 \vee \overline{x_6})$$

8

to

$$C' = (x_1 \vee \overline{y_1}) \wedge (y_1 \vee \overline{x_3} \vee \overline{y_2}) \wedge (y_2 \vee \overline{x_4} \vee \overline{y_3}) \wedge (y_3 \vee x_5 \vee \overline{y_4}) \wedge (y_4 \vee \overline{x_6} \vee \overline{y_5}) \wedge y_5$$

Problem $\sharp$ 4.

1. Define the 0/1 Integer Programming problem (**0/1 IP**), and describe a polynomial-time reduction from **3SAT** to **0/1 IP**.
2. Based on the above reduction construct a **0/1 IP** instance for the following **3SAT** instance:

$$(\neg x_1 \vee x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_2 \vee x_4)$$

For the Boolean assignment $x_1 = 0, x_2 = 1, x_3 = 0, x_4 = 1$ check if the inequalities in the corresponding instance of **0/1 IP** are satisfied. (Explicitly show the inequalities!)

(1) Input. An integer matrix C and an integer vector d

Question Is there a 0/1 vector c such that $C.c \geq d$?

5

3SAT $\leq_p$ 0/1 IP. Let F be a 3-CNF formula

$F = F_1 \wedge F_2 \wedge \ldots \wedge F_k$ with var. $x_1, \ldots, x_n$

10

Define $C = (c_{ij})$ $1 \leq i \leq k$, $1 \leq j \leq n$ by

$$c_{ij} := \begin{cases} 1 & \text{if } x_j \text{ occurs in } F_i \\ -1 & \text{if } \overline{x_j} \text{ ———— } F_i \\ 0 & \text{otherwise} \end{cases}$$

$d_i := 1 - \#$ of var. x s.t. $\overline{x}$ occurs in F_i

(2)

$$C = \begin{bmatrix} -1 & 1 & -1 & 0 \\ 1 & -1 & -1 & 0 \\ 1 & -1 & 0 & 1 \end{bmatrix} \qquad d = \begin{bmatrix} 1-2 \\ 1-2 \\ 1-1 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix}$$

10

$$\left.\begin{aligned} x_1 &= 0 \\ x_2 &= 1 \\ x_3 &= 0 \\ x_4 &= 1 \end{aligned}\right\} \Rightarrow c = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \Rightarrow C \times c = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \geq d$$

$\Longrightarrow$ c is a solution