COL351 Quiz 3

Viraj Agashe

TOTAL POINTS

7.5 / 10

QUESTION 1

1 Q1 5 / 5

✓ + **5 pts** **Correct**

+ **0 pts** Incorrect

+ **4 pts** Algorithm + Reasoning

+ **1 pts** O(n*Log(n) Time Complexity Reasoning

QUESTION 2

2 Q2 2.5 / 5

+ **0 pts** Incorrect

2(a)

✓ + **2.5 pts** **2(a) Correct**

+ **0 pts** 2(a) Incorrect

+ **2 pts** 2(a) 10 Iteration of FF Algorithm

+ **0.5 pts** 2(a) Time Complexity and Reasoning

2(b)

+ **5 pts** 2(b) Correct

+ **0 pts** 2(b) Incorrect

+ **2.5 pts** 2(b) Unique min-cut if S,T forms a partition of V

+ **2 pts** 2(b) Correctness

+ **0.5 pts** 2(b) Time Complexity and Reasoning

+ **1 pts** 2(b) Partial Correctness

1 10 iterations are required.

1 Q1 5 / 5

- ✓ + **5 pts** Correct
- + **0 pts** Incorrect
- + **4 pts** Algorithm + Reasoning
- + **1 pts** O(n*Log(n) Time Complexity Reasoning

2 Q2 **2.5 / 5**

+ **0 pts** Incorrect

2(a)

✓ + **2.5 pts** **2(a) Correct**

+ **0 pts** 2(a) Incorrect

+ **2 pts** 2(a) 10 Iteration of FF Algorithm

+ **0.5 pts** 2(a) Time Complexity and Reasoning

2(b)

+ **5 pts** 2(b) Correct

+ **0 pts** 2(b) Incorrect

+ **2.5 pts** 2(b) Unique min-cut if S,T forms a partition of V

+ **2 pts** 2(b) Correctness

+ **0.5 pts** 2(b) Time Complexity and Reasoning

+ **1 pts** 2(b) Partial Correctness

1 10 iterations are required.

COL351: Quiz-3 Name: VIRAJ AGASHE

Maximum marks: 5+5 = 10 Entry number: | 2 | 0 | 2 | 0 | C | S | 1 | 0 | 5 | 6 | 7 |

Question 1 Let A and B be two sets, each having n integers in the range $[1, 10n]$. The Cartesian sum of A and B is $C = \{x+y \mid x \in A,\ y \in B\}$. We want to find the elements of C and the number of times each element of C is realized as a sum of elements in A and B. Design an $O(n \log n)$ time algorithm to achieve this objective [5 marks].

Sol. For each $a_i \in A$ construct a polynomial,

$$A(x) = x^{a_0} + x^{a_1} + \cdots + x^{a_{n-1}}$$

Similiarly for each $b_i \in B$,

$$B(x) = x^{b_0} + x^{b_1} + \cdots + x^{b_{n-1}}.$$

Then note that

$$C(x) = A(x) \cdot B(x) = \sum_{\cancel{a+b}\ k} c_k \cdot x^{a+b} \quad \text{for } a \in A, b \in B.$$

$\therefore$ The powers of the variable x in the polynomial $C(x) = A(x) \cdot B(x)$ gives the possible values of ~~any c for the~~ $a+b$ for $a \in A$ and $b \in B$.

Further, the coefficients of x^c in $C(x)$ gives the no. of times x^c is realized as a multiplication of $x^a \cdot x^b$, i.e. the no. of times we get $c = a+b$.

Hence, we can multiply the 2 polynomials $A(x)$ & $B(x)$ using polynomial multiplication algorithm using FFT which is $O(n \log n)$.

Question 2 Let G be a directed graph with unit edge capacities, s be a source, t be a destination. Present a linear time algorithm to verify if (s, t)-max flow in G is bounded by nine [2.5 marks].

OR

Let G be a directed graph with unit edge capacities, s be a source, t be a destination. Design an $O(mn)$ time algorithm to verify if G has a unique (s, t)-min-cut [5 marks].

$mn(m+n)$

$(m+n).|F|$.

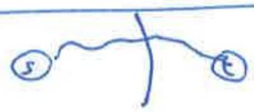
Sol. Part 1.

Consider this

We run iterations of the Ford-Fulkerson algorithm. ~~Since capacities are integers, the flow increases in each iteration~~

We know that since capacities are ints ≥ 1, the iterations taken by the algo is $|F|$. Therefore, we can run 9 iterations of Ford-Fulkerson. If we can still ~~increase~~ modify the value of the max flow (i.e. path from $s-t$ exists) after 9 iterations, then clearly $|F| > 9$. Otherwise, $|F| \leq 9$.

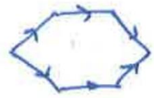
This is a $O((m+n)|F|) = O(9(m+n))$
$= O(m+n)$
algorithm. (linear)



~~Part 2: This is equivalent to checking that~~

Part 2. We can ~~check~~ find using Ford-Fulkerson in $O((m+n)|F|)$ time, the max flow in the graph. Note that since all wts are 1, $|F| \leq deg(s) \leq n$.

X



$\therefore$ In $O(mn)$ time we can find max flow.

~~Now, we can remove edges from the flow and run~~