CS 6363: Design and Analysis of Algorithms
Exam #2, November 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: 10.00 – 11.20 am

Good Luck!

#1	#2	#3	#4	Total

Problem ♯ 1. (Matroid and Graphic Matroid)

1. Define the notions of an independent system and a matroid.

Let S be a finite set, $\mathbb{I}$ be a nonempty family of subsets of S.

- $(S, \mathbb{I})$ is an independent system if (hereditary)

$A \subseteq B \wedge B \in \mathbb{I} \implies A \in \mathbb{I}$

10

- An independent system $(S, \mathbb{I})$ is a matroid if it sat.

$A, B \in \mathbb{I} \wedge |A| > |B| \implies \exists x \in A - B : B \cup \{x\} \in \mathbb{I}$

(which is called the exchange property)

2. (Graphic Matroid) Let $G = (V, E)$ be an undirected graph. Let $\underset{\sim}{S} = E$ and $\underset{\sim}{I}$ be the collection of sets of edges in $\underset{\sim}{S}$ such that each set induces a forest. Show that $(\underset{\sim}{S}, \underset{\sim}{I})$ is a matroid.

(a) $(S, \mathbb{I})$ is an indep. system:

$A \subseteq B \wedge B \in \mathbb{I} \implies B$ induces a forest

$A \subseteq B \implies A$ induces a forest $\implies A \in \mathbb{I}$

(b) Let $A, B \in \mathbb{I}$ and $|A| > |B|$.

A, B induce forests. $|A| > |B| \implies$ There are more trees in B than A.

15

$\implies$ Some tree in A has vertices in two different trees in B

$\implies \exists$ edge $e \in A$ connecting these trees

$\implies B \cup \{e\}$ is cycle free

$\implies B \cup \{e\}$ induces a forest

$\implies B \cup \{e\} \in \mathbb{I}$.

Problem ♯ 2. (Dijkstra's algorithm) Let $G = (V, E)$ be a directed graph with a weight function $w : E \longrightarrow \mathbf{R}_+$

1. Describe what it means to relax an edge $e = (u, v)$

RELAX (u,v) :
 if $d[v] > d[u] + w[u,v]$
  then $d[v] := d[u] + w[u,v]$
   $\pi[v] := u$

4

2. Describe Dijkstra's algorithm, and analyze its running time.

$S = \{s\}$ /* s = source */ Initialize (G, s)
$Q = V - \{s\}$
while $Q \neq \phi$ do
 u = EXTRACT-MIN(Q)
 $S = S \cup \{u\}$
 for each $v \in Adj[u]$ do
  RELAX (u,v)

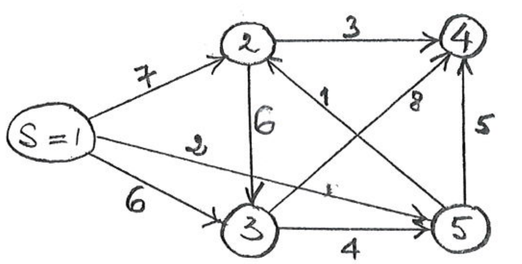
6

- Constr. min heap: $O(n)$
- RELAX (u,v) is performed $O(E)$ times, each requires $O(\log n)$ to maintain min heap

$\Rightarrow$ Total is $O(E \log n)$

5

3. Perform Dijkstra algorithm on the folllowing graph:



10

Step	u	Q	d
0		{2,3,4,5}	$[7,6,\infty,\cancel{2}]$
1	5	{2,3,4}	$[\cancel{3},6,7,\cancel{2}]$
2	2	{3,4}	$[\cancel{3},\cancel{6},6,\cancel{2}]$
3	3	{4}	$[\cancel{3},\cancel{6},\cancel{6},\cancel{2}]$
4	4	{ }	

Problem ♯ 3. (Pattern Matching)

1. Let Σ be an alphabet. For a given pattern $P[1..m]$ define the *suffix* function $\sigma : \Sigma^* \longrightarrow \{0, ..., m\}$ and the *prefix* function $\pi : \{1, ..., m\} \longrightarrow \{0, ..., m-1\}$.

3 Suffix function: $\sigma(x) = \max\{k \mid P_k \sqsupset x\}$

3 Prefix function: $\pi[q] = \max\{k < q \mid P_k \sqsupset P_q\}$

2. Describe the KMP algorithm for pattern matching where the text $T[1..n]$ and the pattern $P[1..m]$ are in the input. (You may assume that π is given.) Analyze the running time of your algorithm.

KMP_Matcher (T, n, P, m)

```
q = 0
for i = 1 to n do
    while q > 0 and P[q+1] ≠ T[i]
        do q = π(q);                    (*)
    if P[q+1] = T[i] then q = q+1       (**)
    if q = m then
        print "i-m is valid shift"
        q = π(q)
```

6

5

(**) ⇒ is performed at most O(n) times

⇒ (*) is performed at most O(n) times

⇒ Total running time is O(n)

3. Perform the KMP algorithm for the following input: (Show the steps of your computation!)

P	a	b	a	b	a
k	1	2	3	4	5
π	0	0	1	2	3

8

T[1..9]

				5	6	7	8	9
a	a	a	b	a	b	a	b	a

q = 0

a	b		

$\underline{i=1}$: $P[1] = T[1]$ $\Rightarrow q = q+1 = 1$

$\underline{i=2}$: $P[2] \neq T[2]$ $\Rightarrow q = \pi(1) = 0$

$\underline{i=3}$: $P[1] = T[3]$ $\Rightarrow q = q+1 = 1$

$\underline{i=4}$: $P[2] = T[4]$ $\Rightarrow q = q+1 = 2$

$\underline{i=5}$ $P[3] = T[5]$ $\Rightarrow q = q+1 = 3$

$\underline{i=6}$ $P[4] = T[6]$ $\Rightarrow q = q+1 = 4$

$\underline{i=7}$ $P[5] = T[7]$ $\Rightarrow q = q+1 = 5$

$q = m = 5$: "2 = i-m is valid shift"

$q = \pi(5) = 3$

$\underline{i=8}$ $P[4] = T[8]$ $\Rightarrow q = q+1 = 4$

$\underline{i=9}$ $P[5] = T[9]$ $\Rightarrow q = q+1 = 5$

a	b	a	b	a

$q = m = 5$

"4 is valid shift"

i=8

a	b	a	

Problem ♯ 4. (Network Flows) Let $G = (V, E)$ be a directed graph together with a capacity function $c : V \times V \longrightarrow \mathbf{R}$ which forms a flow network.

1. Define the *residual network* G_f for a given flow f. Define the *flow value* $\| f \|$ of f. For a cut (S, T) define the *net flow across the cut* (S, T) and the capacity $c(S, T)$ of the cut (S, T). Argue that $\| f \| \leq c(S, T)$.

4

Residual capacity of (u,v) is $c_f(u,v) = c(u,v) - f(u,v)$

Residual NW: $G_f = (V, E_f)$ where $E_f = \{(u,v) \in V \times V \mid c_f(u,v) > 0\}$

Netflow across (S,T) is $f(S,T) = \sum_{u \in S, v \in T} f(u,v)$

Capacity of (S,T) is $c(S,T) = \sum_{u \in S, v \in T} c(u,v)$

4

Claim. $\|f\| \leq c(S,T)$

Pf. For $u \in S$, $v \in T$: $f(u,v) \leq c(u,v)$ due to def. of flow. Thus,

$$\|f\| \overset{\text{Lem.}}{=} f(S,T) = \sum_{u \in S, v \in T} f(u,v) \leq \sum_{u \in S, v \in T} c(u,v) = c(S,T)$$

2. Desribe Edmonds-Karp algorithm and state its running time.

6

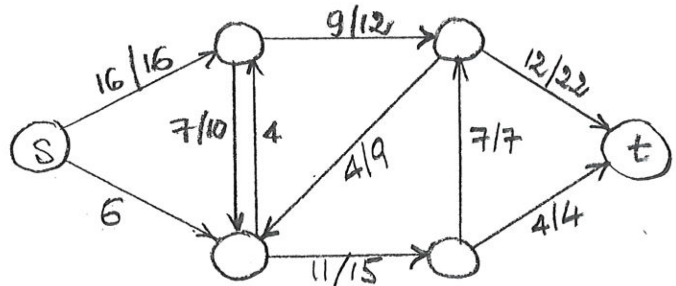
Edmonds-Karp (G, c, s, t)
for all $u, v \in V$ do $f(u,v) = 0$;
$G_f = G$;
while $\exists$ augm. path p obtained by BFS in G_f
do augm. f along p
update f
return f

Complexity:

3 $O(VE^2)$

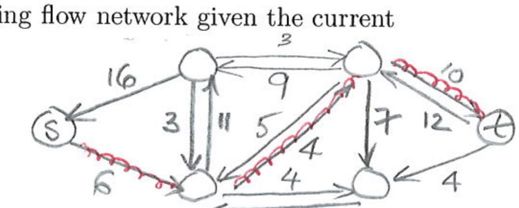
since loop is excecuted $O(VE)$ times

3. Perform the Edmonds-Karp algorithm on the following flow network given the current flow: (Show a min cut!)

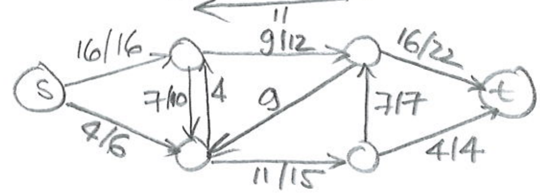


Step 1:

G_f:



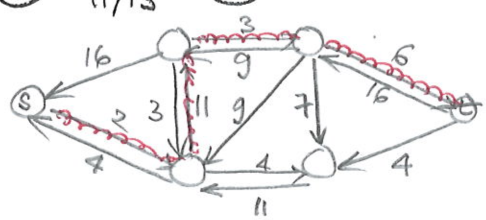
New flow:



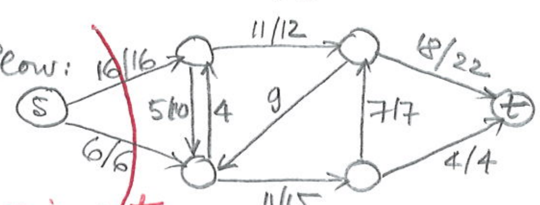
8

Step 2.

G_f



New flow:



max flow = 22, min cut