Math 482 Linear programming
Fall 2020
Final exam
12/14/2020
Time Limit: 180 Minutes

Name (Print): Wenxiao Yang

Section: C13

This is the 3 CREDIT EXAM.

This exam contains 8 pages (including this cover page) and 7 problems. Check to see if any pages are missing. Enter all requested information on the top of this page.

You may use your books, notes, or any calculator on this exam.

You are required to justify your work on each problem on this exam. The following rules apply:

- **Organize your work**, in a reasonably neat and coherent way, in the space provided. Work scattered all over the page without a clear ordering will receive very little credit.
- **Mysterious or unsupported answers will not receive full credit**. A correct answer, unsupported by calculations, explanation, or algebraic work will receive no credit; an incorrect answer supported by substantially correct calculations and explanations might still receive partial credit.

Do not write in the table to the right.

Problem	Points	Score
1	14	
2	18	
3	12	
4	12	
5	16	
6	14	
7	14	
Total:	100	

1. Consider the linear program below.

$$\begin{aligned} \underset{x,y\in\mathbb{R}}{\text{minimize}} \quad & 6x + y \\ \text{subject to} \quad & x \geq 1, \\ & -3x + 4y \geq 1, \\ & 2x - y = 1, \\ & x \text{ unconstrained}, y \geq 0. \end{aligned}$$

(a) (6 points) Write down the dual of the linear program.

$$\begin{aligned} \max_{u\in\mathbb{R}^3} \quad & u_1 + u_2 + u_3 \\ \text{s.t.} \quad & u_1 - 3u_2 + 2u_3 = 6 \\ & 4u_2 - u_3 \leq 1 \\ & u_1, u_2 \geq 0 \\ & u_3 \text{ unconstrained.} \end{aligned}$$

(b) (8 points) If $(x, y) = (1, 1)$ is the optimal solution to the linear program above, use complementary slackness to find a dual optimal solution.

when $(x,y) = (1,1)$

$\begin{cases} x = 1 \\ -3x + 4y = 1 \\ 2x - y = 1 \end{cases} \Rightarrow$ all constraints in (P) are tight.

$\begin{cases} x = 1 \\ y = 1 \end{cases} \Rightarrow \begin{cases} u_1 - 3u_2 + 2u_3 = 6 \\ 4u_2 - u_3 = 1 \end{cases} \Rightarrow (u_1, u_2, u_3) = (8 - 5u_2, u_2, 4u_2 - 1)$

Let $(u_1, u_2, u_3) = (8, 0, -1)$ is dual feasible

and $(x, y) = (1, 1)$ is primal feasible

all constraints in (D) and (P) are tight

$\Rightarrow$ they satisfy complementary slackness

$\Rightarrow$ they are optimal

$\Rightarrow (8, 0, -1)$ is dual optimal

2. (18 points) Consider the primal-dual pair below.

$$(P)\begin{cases}\underset{\mathbf{x}\in\mathbb{R}^4}{\text{minimize}} & x_1+2x_2+3x_3+4x_4\\ \text{subject to} & x_1-3x_2+3x_3+x_4=2\\ & -x_1+4x_2+2x_3+x_4=5\\ & x_1,x_2,x_3,x_4\geq 0\end{cases}\qquad (D)\begin{cases}\underset{u_1,u_2\in\mathbb{R}}{\text{maximize}} & 2u_1+5u_2\\ \text{subject to} & u_1-\ u_2\leq 1\\ & -3u_1+4u_2\leq 2\\ & 3u_1+2u_2\leq 3\\ & u_1+\ u_2\leq 4\end{cases}$$

Starting from the dual solution $u=(u_1,u_2)=(1,0)$, perform **one iteration** of the primal-dual algorithm, updating u to a new dual solution.

$$(DRP):\ \begin{array}{ll}\underset{V\in\mathbb{R}^2}{\max} & 2V_1+5V_2\\ & V_1-V_2\leq 0\\ & 3V_1+2V_2\leq 0\\ & V_1\leq 1\\ & V_2\leq 1\end{array}\qquad (RP):\ \begin{array}{ll}\underset{y\in\mathbb{R}^2}{\min} & y_1+y_2\\ & x_1+3x_3+y_1=2\\ & -x_1+2x_3+y_2=5\\ & x_1,x_3,y_1,y_2\geq 0\end{array}$$

	x_1	x_3	y_1	y_2	
y_1	1	3	1	0	2
y_2	-1	2	0	1	5
$-z$	0	0	1	1	0

$\longrightarrow$

	x_1	x_3	y_1	y_2	
y_1	1	3	1	0	2
y_2	-1	2	0	1	5
$-z$	0	-5	0	0	-7

pivot x_3 $\longrightarrow$ remove y_1

	x_1	x_3	y_1	y_2	
x_3	$\frac{1}{3}$	1	$\frac{1}{3}$	0	$\frac{2}{3}$
y_2	$-\frac{5}{3}$	0	$-\frac{2}{3}$	1	$\frac{11}{3}$
$-z$	$\frac{5}{3}$	0	$\frac{5}{3}$	0	$-\frac{11}{3}$

$$V^T=1-r_y^T=\left(-\frac{2}{3},1\right)$$

$$u=\left(1-\frac{2}{3}t,\ t\right)\quad\begin{cases}\left(1-\frac{2}{3}t\right)-t\leq 1\\ -3+2t+4t\leq 2\\ 1-\frac{2}{3}t+t\leq 4\end{cases}\Rightarrow t\leq\frac{5}{6}\Rightarrow t=\frac{5}{6}$$

$$\Longrightarrow \text{new } u=\left(\frac{4}{9},\frac{5}{6}\right)$$

3. (12 points) Alice and Bob each have three different poker chips: one worth \$10, one worth \$20, and one worth \$30. They each simultaneously put down a poker chip. If their poker chips have the same value, Alice takes Bob's chip. Otherwise, Bob takes Alice's chip. Draw a payoff matrix for this game and write down (but do not solve) a linear program by which Alice could determine her optimal strategy.

	B_{10}	B_{20}	B_{30}
A_{10}	10	-10	-10
A_{20}	-20	20	-20
A_{30}	-30	-30	30

$$\max \ u.$$

$x \in \mathbb{R}^3 \ u \in \mathbb{R}$

$$x_1 + x_2 + x_3 = 1.$$

$$u \leq 10x_1 - 20x_2 - 30x_3$$

$$u \leq -10x_1 + 20x_2 - 30x_3$$

$$u \leq -10x_1 - 20x_2 + 30x_3$$

$$x_1, x_2, x_3 \geq 0$$

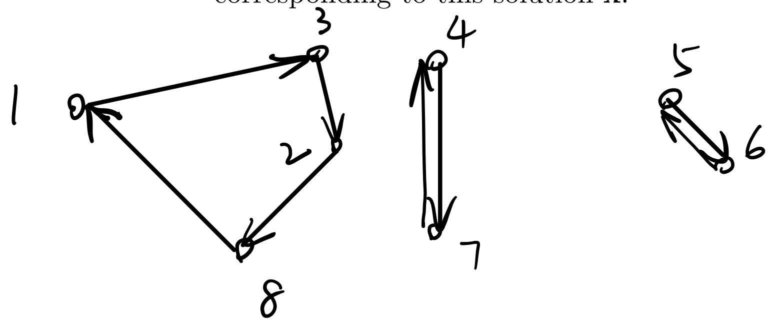
u unrestricted.

4. Suppose that we are solving an instance of the traveling salesperson problem with 8 cities. We get an intermediate solution $\mathbf{x}$ in which

$$x_{13} = x_{28} = x_{32} = x_{47} = x_{56} = x_{65} = x_{74} = x_{81} = 1$$

and all other variables are 0.

(a) (6 points) Draw a diagram of the collection of subtours (it's not a single complete tour) corresponding to this solution $\mathbf{x}$.



(b) (6 points) Find two different subtour elimination constraints violated by this solution. Write them down in full (without $\sum$ notation).

$$\sum_{i \in \{5,6\}} \sum_{j \notin \{5,6\}} x_{ij} \geq 1.$$

$$= x_{51} + x_{52} + x_{53} + x_{54} + x_{57} + x_{58}$$
$$+ x_{61} + x_{62} + x_{63} + x_{64} + x_{67} + x_{68} \geq 1.$$

$$\sum_{i \in \{4,7\}} \sum_{j \notin \{4,7\}} x_{ij} \geq 1$$

$$= x_{41} + x_{42} + x_{43} + x_{45} + x_{46} + x_{48}$$
$$+ x_{71} + x_{72} + x_{73} + x_{75} + x_{76} + x_{78} \geq 1.$$

5. (16 points) Consider the integer program below.

$$\begin{aligned} \underset{x,y\in\mathbb{Z}}{\text{maximize}} \quad & x+y \\ \text{subject to} \quad & 2y \le 2x+1 \\ & y \ge 4x-5 \\ & x,y \ge 0 \end{aligned}$$

Perform **one iteration** of the cutting plane method on this integer program. This means you need to add one fractional cut, and then once you have added the cut, use dual simplex method to make your tableau dual feasible.

$\max\ x+y$

$x,y\in\mathbb{Z}$

s.t. $-2x+2y+s_1\le 1$

$4x-y+s_2\le 5$

$x,y,s_1,s_2\ge 0$

	x	y	s_1	s_2	
s_1	-2	2	1	0	1
s_2	4	-1	0	1	5
$-z$	1	1	0	0	0

$\xrightarrow[\text{remove } s_2]{\text{pivot } x}$

	x	y	s_1	s_2	
s_1	0	$\frac{3}{2}$	1	$\frac{1}{2}$	$\frac{7}{2}$
x	1	$-\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{5}{4}$
$-z$	0	$\frac{5}{4}$	0	$-\frac{1}{4}$	$-\frac{5}{4}$

$\xrightarrow[\text{remove } s_1]{\text{pivot } y}$

	x	y	s_1	s_2	
y	0	1	$\frac{2}{3}$	$\frac{1}{3}$	$\frac{7}{3}$
x	1	0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{11}{6}$
$-z$	0	0	$-\frac{5}{6}$	$-\frac{2}{3}$	$-\frac{25}{6}$

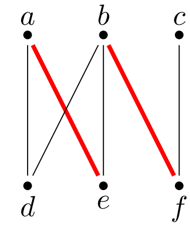
$y+\frac{2}{3}s_1+\frac{1}{3}s_2=\frac{7}{3} \Rightarrow y\le 2 \quad \frac{2}{3}s_1+\frac{1}{3}s_2\ge\frac{1}{3}$

	x	y	s_1	s_2	s_3	
y	0	1	$\frac{2}{3}$	$\frac{1}{3}$	0	$\frac{7}{3}$
x	1	0	$\frac{1}{6}$	$\frac{1}{3}$	0	$\frac{11}{6}$
s_3	0	0	$-\frac{2}{3}$	$-\frac{1}{3}$	1	$-\frac{1}{3}$
$-z$	0	0	$-\frac{5}{6}$	$-\frac{2}{3}$	0	$-\frac{25}{6}$

$\xrightarrow[\text{remove } s_3]{\text{pivot } s_1}$

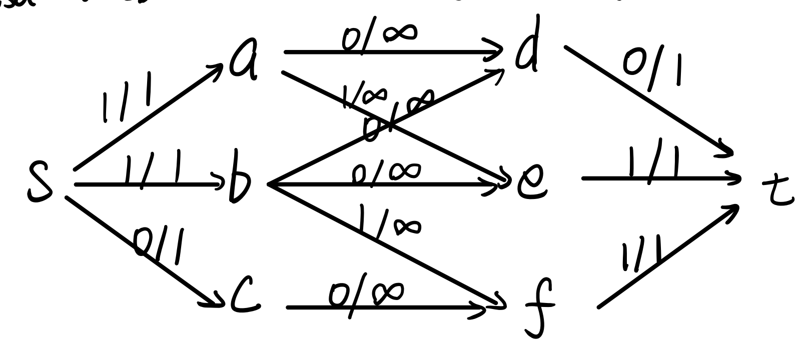
	x	y	s_1	s_2	s_3	
y	0	1	0	0	1	2
x	1	0	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{7}{4}$
s_3	0	0	1	$\frac{1}{2}$	$-\frac{3}{2}$	$\frac{1}{2}$
$-z$	0	0	0	$-\frac{1}{4}$	$-\frac{5}{12}$	$-\frac{15}{4}$

6. (14 points) Consider the following matching (that is, $M = \{(a,e),(b,f)\}$) in a bipartite graph:

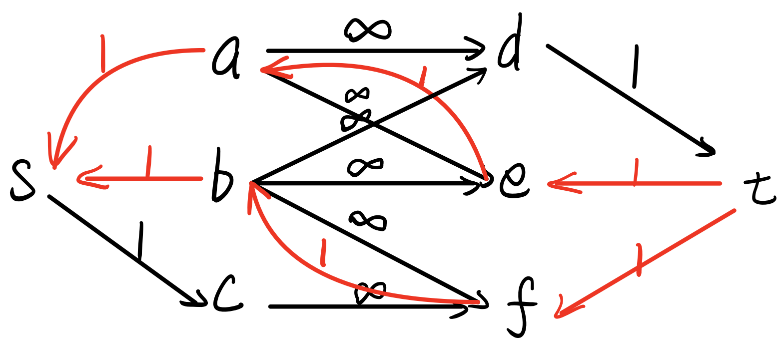


First, convert this matching into a feasible flow in a network. Then, use the Ford-Fulkerson algorithm to find the max flow starting with this feasible flow (make sure you **draw the residual graph** and write out the **augmenting path**). Then convert the max flow to a maximal matching.

$\{x_{sa} = x_{sb} = x_{ae} = x_{bf} = x_{et} = x_{ft} = 1,\ x_{sc} = x_{ad} = x_{bd} = x_{be} = x_{cf} = x_{dt} = 0\}$



residual graph:



go: $s \to c \to f \to b \to d \to t$

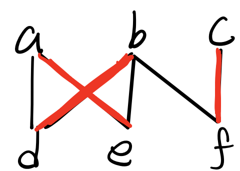
augmenting path: $s \xrightarrow{0/1} c \xrightarrow{0/\infty} f \xleftarrow{1/\infty} b \xrightarrow{0/\infty} d \xrightarrow{0/1} t$

$\xRightarrow{\text{adding}}$ 1 $\quad s \xrightarrow{1/1} c \xrightarrow{1/\infty} f \xleftarrow{0/\infty} b \xrightarrow{1/\infty} d \xrightarrow{1/1} t$

max flow is $\underline{3}$.

$\{x_{sa} = x_{sb} = x_{ae} = x_{sc} = x_{et} = x_{ft} = x_{bd} = x_{cf} = 1,\ x_{ad} = x_{bf} = x_{be} = 0\}$

$\Rightarrow$ max matching:



$M = \{(c,f),(b,d),(a,e)\}.$

7. (14 points) One of the consequences of the Four-Color Theorem, a famous result in combinatorics proved here at UIUC, is that a map of the 48 continental states of the United States can be colored with 4 colors in such a way that adjacent states receive different colors.

   Describe how to write an integer program to find such a coloring of the states, using the colors red, blue, yellow, and green, so that green is used to color as few states as possible.

   (There will be many similar-looking constraints, and they depend on the adjacencies between states, which I don't expect you to have memorized. It's fine if you give an example of each type of constraint, and then ***explain in words, using full sentences, how that example generalizes****. To help you do this: the state of Illinois is adjacent to the states Michigan, Wisconsin, Indiana, Kentucky, Missouri, and Iowa.)*

$k = \{\text{red}, \text{blue}, \text{yellow}, \text{green}\}$. $S = \{48 \text{ states in US}\}$.

$$x_{ij} = \begin{cases} 1, & \text{if state } i \text{ is colored by } j. \\ 0, & \text{otherwise}. \end{cases}$$

$A = \{ (i,t) \mid \text{state } i \text{ is adjacent to state } k, \ i \in S, k \in S \}$.

$$\min_{x \in \mathbb{Z}^{192}} \sum_{i \in S} x_{i\,\text{green}}$$

s.t. $\sum_{j \in k} x_{ij} = 1$, $\forall i \in S$: every state is colored by one color.

$x_{ij} + x_{tj} \le 1$, $\forall (i,t) \in A$, $\forall j \in k$: state i and state k are adjacent, so they don't have same color.

$x_{ij} \in \{0,1\}$ $\forall i \in S$, $\forall j \in k$