**ISE-DSA 5113 Homework 3**

**Team Members:**

* Soumya Sri Kesani
* Orkhan Khankishiyev
* Tahsin Tabassum
* Anvitha Reddy Thummalapally

**Q1**

1. **Solution:**

Formulation of the problem as MCNFP:

Considering 6 English and French children, the graph looks like this:

A black and white image of a network

Description automatically generated

The left-side nodes indicate English children, and the right-side nodes indicate French children. The left-side nodes will have a “b” value of 1, and the right-side nodes will have a “b” value of -1. The cost of the arcs represents their willingness to form pairs with the receiving node. The lower the value, the more they are compatible with each other.

The problem was solved in AMPL, creating a small example instance. The data file and solution of the problem are here:

**Code**

reset;

options solver cplex;

set NODES; # nodes in the network

set ARCS within {NODES, NODES}; # arcs in the network

param b {NODES} default 0; # supply/demand for node i

param c {ARCS} default 0; # cost of one of flow on arc(i,j)

param l {ARCS} default 0; # lower bound on flow on arc(i,j)

param u {ARCS} default Infinity; # upper bound on flow on arc(i,j)

var x {ARCS}; # flow on arc (i,j)

minimize cost: sum{(i,j) in ARCS} c[i,j] \* x[i,j]; #objective: minimize arc flow cost

# Flow Out(i) - Flow In(i) = b(i)

subject to flow\_balance {i in NODES}:

sum{j in NODES: (i,j) in ARCS} x[i,j] - sum{j in NODES: (j,i) in ARCS} x[j,i] = b[i];

subject to capacity {(i,j) in ARCS}: l[i,j] <= x[i,j] <= u[i,j];

#data

Data 1a.dat;

#solve

solve;

#for {(i,j) in ARCS} {

# printf i,j, X[i,j];

#}

display x;

**Data file:**

data;

set NODES := E1 E2 E3 E4 E5 E6 #English children

F1 F2 F3 F4 F5 F6; #French members

#Arcs go from English members to French members

set ARCS := (E1, \*) F1 F2 F3 F4 F5 F6

(E2, \*) F1 F2 F3 F4 F5 F6

(E3, \*) F1 F2 F3 F4 F5 F6

(E4, \*) F1 F2 F3 F4 F5 F6

(E5, \*) F1 F2 F3 F4 F5 F6

(E6, \*) F1 F2 F3 F4 F5 F6;

param b:=

E1 1

E2 1

E3 1

E4 1

E5 1

E6 1

F1 -1

F2 -1

F3 -1

F4 -1

F5 -1

F6 -1;

param c:= # weights representing willingess to form a pair

[E1, \*] F1 -10 F2 -5 F3 -3 F4 -2 F5 -8 F6 -1

[E2, \*] F1 -1 F2 -10 F3 -8 F4 -5 F5 -2 F6 -6

[E3, \*] F1 -6 F2 -2 F3 -3 F4 -9 F5 -5 F6 -3

[E4, \*] F1 -6 F2 -7 F3 -10 F4 -1 F5 -3 F6 -7

[E5, \*] F1 -3 F2 -4 F3 -5 F4 -8 F5 -10 F6 -1

[E6, \*] F1 -9 F2 -2 F3 -9 F4 -4 F5 -3 F6 -3

;

**Solution obtained from AMPL:**

CPLEX 20.1.0.0: optimal solution; objective -55

1 dual simplex iterations (0 in phase I)

x [\*,\*] :

F1 F2 F3 F4 F5 F6 :=

E1 1 0 0 0 0 0

E2 0 1 0 0 0 0

E3 0 0 0 0 0 0

E4 0 0 0 0 0 0

E5 0 0 0 1 0 0

E6 0 0 1 0 0 0

;

So, the solution dictates that this problem can be solved as an MCNFP. The results show valid pair assignments that maximize total compatibility. And this should be true for all instances of similar kind.

1. **Solution:**

Formulation of the problem as MCNFP:

Considering 6 staff members of the company, the graph looks like this:

A network of black lines and circles

Description automatically generated

The figures should be similar if we consider 16 staff members. The left-side nodes indicate staff members, and the right-side nodes indicate their willingness to form a pair with other members. For instance, an edge of (1-P3) indicates the willingness of member 1 to form a pair with member 3. The question asks for a total of 16 staff members, which is an extension of this graph, where the left-side nodes will have 16 members (1,2,3,4,… …,15,16) and the right-side nodes will have 16 members (P1,P2,P3,P4,… …,P15,P16), and each left-side nodes will have an arc with all right-side nodes expect the representation of its own partner pair. For instance, node 4 will not have an edge with P4. The left-side nodes will have a “b” value of 1, and the right-side nodes will have a “b” value of -1. The cost of the arcs represents their willingness to form pairs with the receiving node. The lower the value, the more they are compatible with each other.

The problem was tried to solve in AMPL, creating a small example instance for 8 people. The data file and solution of the problem are here:

**Data file:**

data;

set NODES := 1 2 3 4 5 6 7 8

P1 P2 P3 P4 P5 P6 P7 P8 ;

set ARCS := (1, \*) P2 P3 P4 P5 P6 P7 P8

(2, \*) P1 P3 P4 P5 P6 P7 P8

(3, \*) P1 P2 P4 P5 P6 P7 P8

(4, \*) P1 P2 P3 P5 P6 P7 P8

(5, \*) P1 P2 P3 P4 P6 P7 P8

(6, \*) P1 P2 P3 P4 P5 P7 P8

(7, \*) P1 P2 P3 P4 P5 P6 P8

(8, \*) P1 P2 P3 P4 P5 P6 P7 ;

param b:=

1 1

2 1

3 1

4 1

5 1

6 1

7 1

8 1

P1 -1

P2 -1

P3 -1

P4 -1

P5 -1

P6 -1

P7 -1

P8 -1 ;

param c:=

[1, \*] P2 8 P3 4 P4 7 P5 3 P6 2 P7 1 P8 5

[2, \*] P1 6 P3 9 P4 2 P5 8 P6 3 P7 7 P8 4

[3, \*] P2 5 P1 3 P4 10 P5 1 P6 6 P7 2 P8 8

[4, \*] P2 9 P3 7 P1 4 P5 5 P6 2 P7 9 P8 3

[5, \*] P2 4 P3 8 P4 6 P1 7 P6 9 P7 3 P8 10

[6, \*] P2 7 P3 2 P4 8 P5 3 P1 1 P7 6 P8 4

[7, \*] P2 3 P3 10 P4 9 P5 2 P6 5 P1 8 P8 7

[8, \*] P2 6 P3 5 P4 7 P5 4 P6 8 P1 3 P7 1;

**Solution obtained from AMPL:**

CPLEX 20.1.0.0: optimal solution; objective 19

9 dual simplex iterations (0 in phase I)

x [\*,\*] :

P1 P2 P3 P4 P5 P6 P7 P8 :=

1 . 0 0 0 1 0 1 0

2 0 . 0 1 0 0 0 0

3 1 0 . 0 0 0 0 0

4 0 1 0 . 0 0 0 0

5 0 1 0 0 . 1 0 0

6 0 0 1 0 0 . 0 0

7 0 0 0 1 0 0 . 0

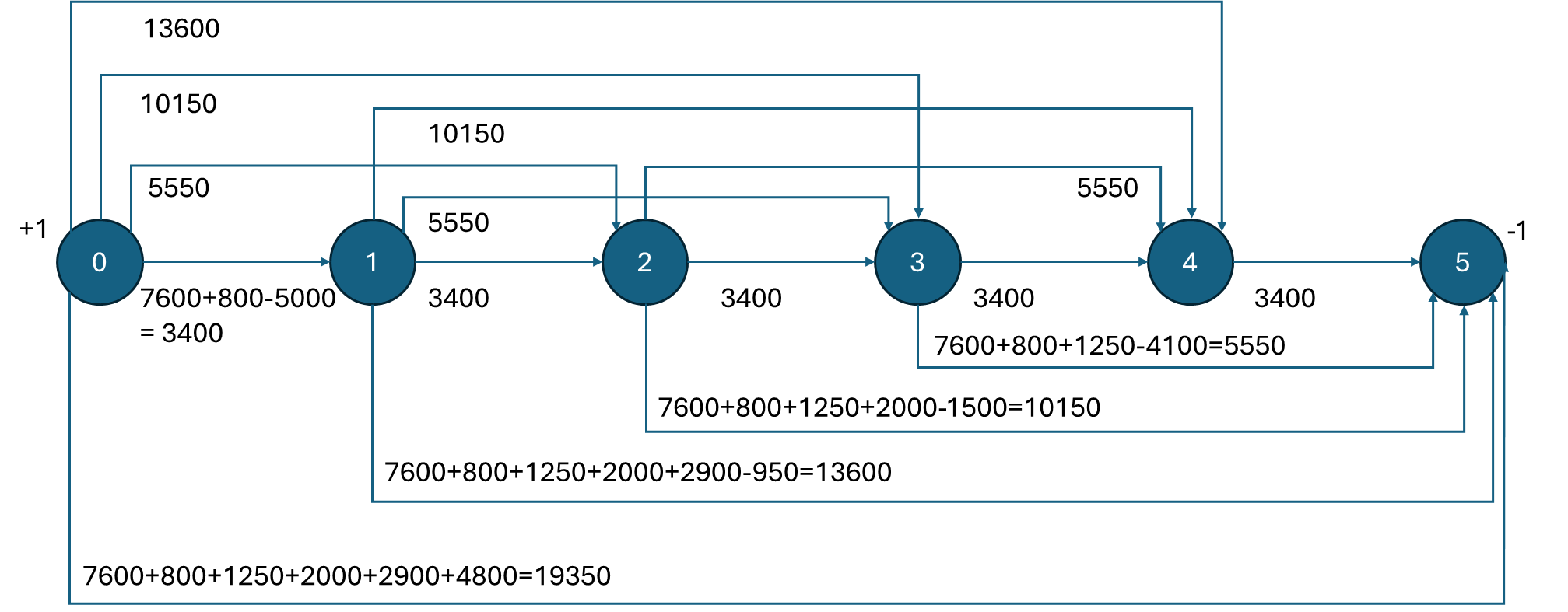
8 0 0 0 0 1 0 0 .

;

It can be seen from the results that the problem does not have valid pair assignments for all the 8 members. Since the previous problem contained different people in supply and demand nodes, it can provide valid pair assignments. But in this case, the demand nodes also represent the same people, and for each person, the “b” values are 1 in the supply nodes and -1 in the demand nodes. So, with the MCNFP approach, valid pair assignments cannot be found. So, this problem cannot be solved with a MCNFP approach.

**Q2**

* The nodes are the years from 0 to 5, where Node 0 has a supply of 1 and Node 5 has a demand of -1.
* The cost for each arc is calculated by considering the cost of buying the grill, the cost of maintenance and the value of selling it on eBay.
* The lower and upper bounds of the arcs are set to 0 and 1, respectively.
* The target is to find the minimum cost network flow using MCNFP code without modifying it. Following diagram is created first.



* The datafile using the above network diagram is as follows:

**set** NODES := 0 1 2 3 4 5;

**set** ARCS := (0, 1) (0, 2) (0, 3) (0, 4) (0, 5) (1, 2) (1, 3) (1, 4) (1, 5) (2, 3) (2, 4) (2, 5) (3, 4) (3, 5) (4, 5);

**param** b :=

0 1

1 0

2 0

3 0

4 0

5 -1;

**param**: c l u :=

[0, 1] 3400 . 1

[0, 2] 5550 . 1

[0, 3] 10150 . 1

[0, 4] 13600 . 1

[0, 5] 19350 . 1

[1, 2] 3400 . 1

[1, 3] 5550 . 1

[1, 4] 10150 . 1

[1, 5] 13600 . 1

[2, 3] 3400 . 1

[2, 4] 5550 . 1

[2, 5] 10150 . 1

[3, 4] 3400 . 1

[3, 5] 5550 . 1

[4, 5] 3400 . 1

;

* This datafile was used to run the MCNFP code provided in the class. The results are as follows:

CPLEX 20.1.0.0: optimal solution; objective 14500

x :=

0 1 1

0 2 0

0 3 0

0 4 0

0 5 0

1 2 0

1 3 1

1 4 0

1 5 0

2 3 0

2 4 0

2 5 0

3 4 0

3 5 1

4 5 0

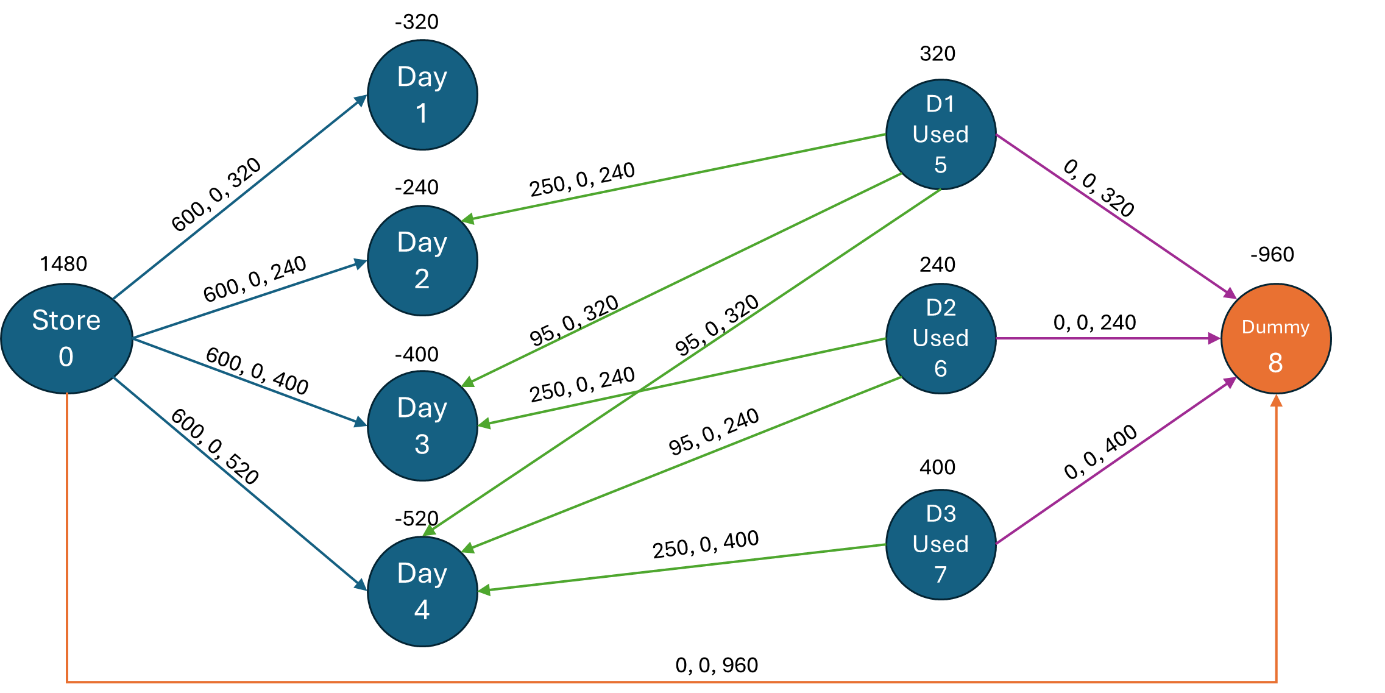
;

cost = 14500

* So, the model suggests maintaining the grill from Year 0 to Year 1. Then sell it on eBay in Year 1 and buy a new grill. Maintain that grill from Year 1 to Year 3, servicing it each year. Then sell it in Year 3 on eBay and buy a new grill once again to keep it till Year 5 by maintaining it each year. This plan should cost a minimum of $14500. Sounds like a really expensive passion.

**Q3.**

* Store (Node 0) is where the team can buy a new tire. The arc going out of the store shows the price of a new tire and the number of tires needed for each day. The total supply from the store is 1480, which is the sum of all tires needed for the race
* There are four race days from day 1 to day 4. The number of tires is shown as demand. The algorithm will consider them as consumed but in reality, they will be used and sent for repair.
* There will be three used tire sets (Node 5, 6 and 7) that can be restored and reused in the upcoming races. The numbers above the nodes shows the supply of tires but they are not the supplier of new tires. That supply corresponds to the number of used tires coming from the race of the day accordingly.
* If overnight restoration is used, the service charge is $250 and if normal restoration is used, the service fee is 95$. The arcs coming out of Used nodes to race day nodes represent the service fees and the number of tires needed for each race day.
* Because there is an overall imbalance of tire supply vs. demand, a dummy demand node was added to fix it. If there is any extra tires left in store or in used service shops, they will flow to the dummy node.
* We are interested in the number of tires going from store to race days and from service shops to race days.



* According to this diagram, the following data file was created to be solved with MCNFP algorithm:

# default node requirements are 0

**set** NODES := 0 1 2 3 4 5 6 7 8;

**set** ARCS := (0, 1) (0, 2) (0, 3) (0, 4) (5, 2) (5, 3) (5, 4) (6, 3) (6, 4) (7, 4)

(5, 8) (6, 8) (7, 8) (0, 8);

**param** b :=

0 1480

1 -320

2 -240

3 -400

4 -520

5 320

6 240

7 400

8 -960

;

**param**: c l u :=

[0, 1] 600 . 320

[0, 2] 600 . 240

[0, 3] 600 . 400

[0, 4] 600 . 520

[5, 2] 250 . 240

[5, 3] 95 . 320

[5, 4] 95 . 320

[6, 3] 250 . 240

[6, 4] 95 . 240

[7, 4] 250 . 400

[5, 8] 0 . 320

[6, 8] 0 . 240

[7, 8] 0 . 400

[0, 8] 0 . 960

;

* And the results obtained are as follows:

CPLEX 20.1.0.0: optimal solution; objective 490000

5 dual simplex iterations (0 in phase I)

x :=

0 1 320

0 2 200

0 3 0

0 4 0

0 8 960

5 2 40

5 3 280

5 4 0

5 8 0

6 3 120

6 4 120

6 8 0

7 4 400

7 8 0

;

cost = 490000

* The results showed that to spend the minimum of $490,000 on tires during 4 race days, we should buy new 320 tires for Day 1, buy new 200 tires and fix 40 used tires overnight for Day 2. We do not buy new tires for Day 3. Instead, we take 280 used tires (from Day 1) fixed in normal service and 120 used tires (from Day 2) fixed overnight. And lasty, for Day 4, we do not buy new tires, but we take 120 used tires (from Day 2) fixed in normal service and 400 used tires (from Day 3) fixed overnight.

**Q4**

Starting Node – A

Specialist – S

Generalist – G

Scranton – SC

Utica – UT

Stamford – ST

Regular Time for Scranton – RT1

Regular Time for Utica– RT2

Regular Time for Stamford– RT3

Over Time for Scranton – OT1

Over Time for Utica– OT2

Over Time for Stamford– OT3

Faceblock – FB (output node)

Goggle – GOG (output node)

Dummy Output Node – OUT

Specialists (S) → Each specialist can produce 12 units per month.

Generalists (G) → Each generalist can produce 10 units per month.

Multipliers:

**A → S (Specialists):** 1/12=0.0833

Each unit of labor sent to S contributes 1/12 of a unit of production capacity.

**A → G (Generalists):** 1/10=0.1

Each unit of labor sent to G contributes 1/10 of a unit of production capacity.

Production Costs:

Each plant has a manufacturing cost per unit:

Scranton: $90 per unit

Utica: $105 per unit

Stamford: $115 per unit

Overtime Costs:

Overtime cost is 1.5 times the regular cost:

Scranton: $90 × 1.5 = $135

Utica: $105 × 1.5 = $157.5

Stamford: $115 × 1.5 = $172.5

Worker transportation cost:

Scranton: $300 per worker

Utica: $250 per worker

Stamford: $275 per worker

A diagram of a network

Description automatically generated

**Data File:**

**data**;

**set** NODES := A S G SC UT ST RT1 OT1 RT2 OT2 RT3 OT3 FB GOG OUT;

**set** ARCS := (A,\*) S G

(S,\*) SC UT ST

(G,\*) SC UT ST

(SC,\*) RT1 OT1

(UT,\*) RT2 OT2

(ST,\*) RT3 OT3

(\*,FB) RT1 OT1 RT2 OT2 RT3 OT3

(\*,GOG) RT1 OT1 RT2 OT2 RT3 OT3

(\*,OUT) FB GOG;

**param**: c:=

[A, \*] S 0 , G 0

[S, \*] SC 2300, UT 2250, ST 2275

[G, \*] SC 2000, UT 1950, ST 1975

[SC, \*] RT1 90, OT1 135

[UT, \*] RT2 105, OT2 157.5

[ST, \*] RT3 115, OT3 172.5

[\*, FB] RT1 15, OT1 15, RT2 18, OT2 18, RT3 20, OT3 20

[\*, GOG] RT1 8, OT1 8, RT2 14, OT2 14, RT3 24, OT3 24

[\*,OUT] GOG -1, FB -1;

**param**: l:=

[A, \*] S 0 , G 0

[S, \*] SC 0, UT 0, ST 0

[G, \*] SC 0, UT 0, ST 0

[SC, \*] RT1 0, OT1 0

[UT, \*] RT2 0, OT2 0

[ST, \*] RT3 0, OT3 0

[\*, FB] RT1 0, OT1 0, RT2 0, OT2 0, RT3 0, OT3 0

[\*, GOG] RT1 0, OT1 0, RT2 0, OT2 0, RT3 0, OT3 0

[\*,OUT] GOG 1000, FB 600;

**param**: u:=

[A, \*] S ., G .

[S, \*] SC 100, UT 100, ST 100

[G, \*] SC 200, UT 200, ST 200

[SC, \*] RT1 505, OT1 100

[UT, \*] RT2 465, OT2 100

[ST, \*] RT3 570, OT3 100

[\*, FB] RT1 505, OT1 100, RT2 505, OT2 100, RT3 505, OT3 100

[\*, GOG] RT1 505, OT1 100, RT2 505, OT2 100, RT3 505, OT3 100

[\*, OUT] GOG 1000, FB 600;

**param**: b:=

A 1600 ;

**param**: mu:=

[A, \*] S 0.0833, G 0.1

[S, \*] SC 12, UT 12, ST 12

[G, \*] SC 10, UT 10, ST 10

[\*, OUT] FB 0, GOG 0;

This data file is used to run the GMCNFP file provided on Canvas.is datafile was used to run the MCNFP code provided in the class. This datafile was used to run the MCNFP code provided in the class. This datafile was used to run the MCNFP code provided in the class.

Output:

A screenshot of a computer

Description automatically generated

A diagram of a network

Description automatically generated

After observing the optimal solution, we can say that, using Generalists would me more profitable. The optimal solution shows that hiring only generalists is the best approach, making certified specialists unnecessary. This decision helps keep costs lower while ensuring a flexible and efficient workforce.

Scranton: 565 generalists (505 working regular time, 60 working overtime)

Utica: 465 generalists (all working regular time)

Stamford: 570 generalists (all working regular time)

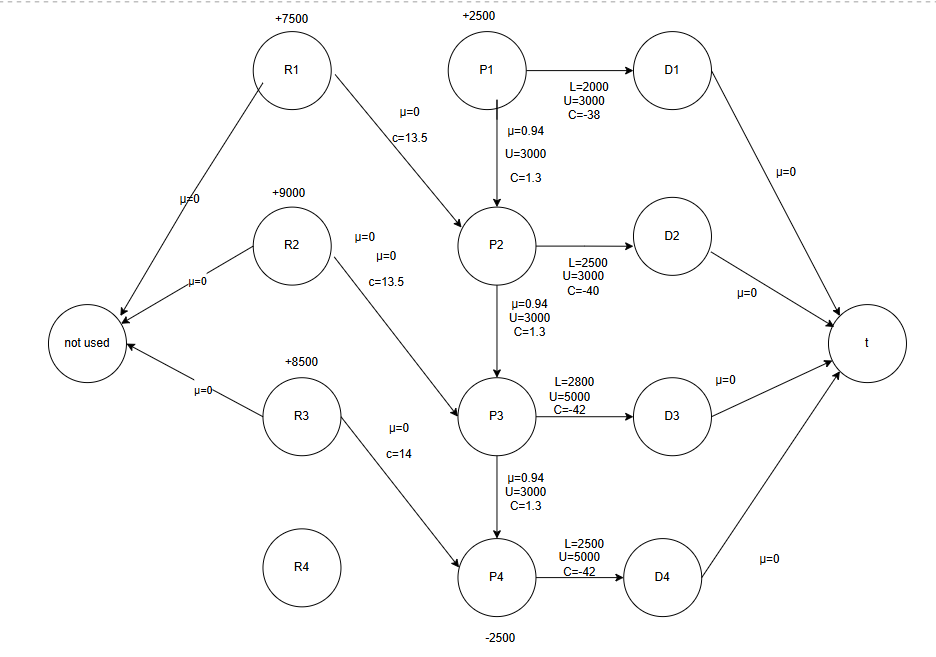
Scranton supplies 565 units to Goggle.

Utica supplies 370 units to Faceblock and 95 units to Goggle.

Stamford supplies 65 units to Faceblock and 505 units to Goggle.

So, Goggle has 1000 units (505+60+370+65) and Faceblock has 600 units (95+505) while achieving the minimum cost of $505,645.

**Q5**



A screenshot of a computer code

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

a)

The optimization problem for MUD-bGone was formulated as a network flow model in AMPL to maximize profit while adhering to constraints on supply, production, storage, and demand. The objective function minimized a negative cost, which effectively maximized the company's total profit.

The optimal solution found for this problem resulted in a maximum profit of **$108,542.50.**

Breakdown for the optimal flow:

1. Production to Demand Fulfillment: The final product MUD-bGone was distributed to meet the demands in each period as follows:

* P1 → D1: 2500 gallons
* P2 → D2: 3000 gallons
* P3 → D3: 3152.5 gallons
* P4 → D4: 2500 gallons

All demand requirements were fully met.

2. Inventory Carried Between Periods:

* P1 → P2: 0 gallons (no inventory carried over)
* P2 → P3: 375 gallons stored and carried to the next period
* P3 → P4: 1250 gallons stored and carried to the next period

This shows that some inventory was stored to manage future demands efficiently while considering storage limits and deterioration.

3. Raw Chemical Base Procurement: The chemical base was procured and used in the following amounts per period:

* R1 → P2: 7500 gallons
* R2 → P3: 9000 gallons
* R3 → P4: 8500 gallons

These values respect the supply constraints and ensure timely processing before meeting demand.

4. Demand Fulfillment to Final Nodes: All demand nodes successfully delivered the required amounts to the terminating node:

* D1 → t: 2500 gallons
* D2 → t: 3000 gallons
* D3 → t: 3152.5 gallons
* D4 → t: 2500 gallons

This confirms that the entire production and distribution process was optimized to fulfill demand while maximizing profit.

b)

The shadow price (dual value) tells us how much profit will increase if we have one more gallon of chemical base available in period 2. In this case, the shadow price for R2 (period 2 supply) is $5.4. This means that for every extra gallon of raw chemical base available in period 2, the profit increases by $5.4. Similarly, if the supply decreases, the profit will drop at the same rate.

This value is only valid within a certain range. The supply of the chemical base in period 2 can increase up to 13,105.6 gallons before the shadow price changes. On the other hand, the supply can decrease down to 8,216.67 gallons before the shadow price is no longer valid.

As long as the supply stays within this range, the profit will continue to increase (or decrease) at a rate of $5.4 per gallon. However, if the supply goes beyond this range, the effect of adding or removing a gallon of supply will no longer be the same, and a new shadow price would apply.

c)

The profit from selling MUD-bGone in period 2 is highly sensitive to changes in its price. If the price drops from $40 to $39, the profit decreases by $3,000. If the price goes further down to $38, the profit decreases by another $2,910.50. This means that even small reductions in price lead to significant losses in profit. While the rate of decrease slows slightly as the price drops further, the overall trend remains the same—lower prices result in lower profits. This shows that the optimal solution is very sensitive to the price of MUD-bGone in period 2, making it a crucial factor in maximizing profit.