Project 3 - Linear Programming

Group 19
John Fitzpatrick, Connor Pacala, Thomas Mathew, Ryan Mackey
July 31, 2016

Problem 1: Transshipment Problem

Problem 1 was completed using LINDO program

pw - plant to warehouse wr - warehouse to retailer

Leading coefficients are cost, trailing coefficients are subscripts in prompt (i, j, k)

Part A - Optimal Solution

Linear program objective function and constraints:

```
MIN 10pw11 + 15pw12 + 11pw21 + 8pw22 + 13pw31 + 8pw32 + 9pw33 + 14pw42 + 8pw43 + 5wr11 + 6wr12 + 7wr13 +
10wr14 + 12wr23 + 8wr24 + 10wr25 + 14wr26 + 14wr34 + 12wr35 + 12wr36 + 6wr37

ST
        pw11 + pw12 < 150
        pw21 + pw22 < 450
        pw31 + pw32 + pw33 < 250
        pw42 + pw43 < 150

        pw11 + pw21 + pw31 - wr11 - wr12 - wr13 - wr14 > 0
        pw12 + pw22 + pw32 + pw42 - wr23 - wr24 - wr25 - wr26 > 0
        pw33 + pw43 - wr34 - wr35 - wr36 - wr37 > 0

        wr11 > 100
        wr12 > 150
        wr13 + wr23 > 100
        wr14 + wr24 + wr34 > 200
        wr25 + wr35 > 200
        wr26 + wr36 > 150
        wr37 > 100

        pw11 > 0
        pw12 > 0
        pw21 > 0
        pw22 > 0
        pw31 > 0
        pw32 > 0
        pw33 > 0
        pw42 > 0
        pw43 > 0
        wr11 > 0
        wr12 > 0
        wr13 > 0
        wr14 > 0
        wr23 > 0
        wr24 > 0
        wr25 > 0
        wr26 > 0
        wr34 > 0
        wr35 > 0
        wr36 > 0
        wr37 > 0
END
```

Optimal solution and output:

```
 LP OPTIMUM FOUND AT STEP     13
        OBJECTIVE FUNCTION VALUE
        1)      17100.00
```

VARIABLE	VALUE	REDUCED COST
PW11	150.000000	0.000000
PW12	0.000000	8.000000
PW21	200.000000	0.000000
PW22	250.000000	0.000000
PW31	0.000000	2.000000
PW32	150.000000	0.000000
PW33	100.000000	0.000000
PW42	0.000000	7.000000
PW43	150.000000	0.000000
WR11	100.000000	0.000000
WR12	150.000000	0.000000
WR13	100.000000	0.000000
WR14	0.000000	5.000000
WR23	0.000000	2.000000
WR24	200.000000	0.000000
WR25	200.000000	0.000000
WR26	0.000000	1.000000
WR34	0.000000	7.000000
WR35	0.000000	3.000000
WR36	150.000000	0.000000
WR37	100.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	1.000000
3)	0.000000	0.000000
4)	0.000000	0.000000
5)	0.000000	1.000000
6)	0.000000	-11.000000
7)	0.000000	-8.000000
8)	0.000000	-9.000000
9)	0.000000	-16.000000
10)	0.000000	-17.000000
11)	0.000000	-18.000000
12)	0.000000	-16.000000
13)	0.000000	-18.000000
14)	0.000000	-21.000000
15)	0.000000	-15.000000
16)	150.000000	0.000000
17)	0.000000	0.000000
18)	200.000000	0.000000
19)	250.000000	0.000000
20)	0.000000	0.000000
21)	150.000000	0.000000
22)	100.000000	0.000000
23)	0.000000	0.000000
24)	150.000000	0.000000
25)	100.000000	0.000000
26)	150.000000	0.000000
27)	100.000000	0.000000
28)	0.000000	0.000000
29)	0.000000	0.000000
30)	200.000000	0.000000
31)	200.000000	0.000000
32)	0.000000	0.000000
33)	0.000000	0.000000
34)	0.000000	0.000000
35)	150.000000	0.000000
36)	100.000000	0.000000

NO. ITERATIONS= 13

RANGES IN WHICH THE BASIS IS UNCHANGED:

OBJ COEFFICIENT RANGES

VARIABLE	CURRENT COEF	ALLOWABLE INCREASE	ALLOWABLE DECREASE
PW11	10.000000	1.000000	INFINITY
PW12	15.000000	INFINITY	8.000000
PW21	11.000000	2.000000	1.000000
PW22	8.000000	5.000000	0.000000
PW31	13.000000	INFINITY	2.000000
PW32	8.000000	0.000000	1.000000
PW33	9.000000	1.000000	1.000000
PW42	14.000000	INFINITY	7.000000
PW43	8.000000	1.000000	INFINITY
WR11	5.000000	INFINITY	16.000000
WR12	6.000000	INFINITY	17.000000
WR13	7.000000	2.000000	18.000000
WR14	10.000000	INFINITY	5.000000
WR23	12.000000	INFINITY	2.000000
WR24	8.000000	5.000000	16.000000
WR25	10.000000	3.000000	18.000000
WR26	14.000000	INFINITY	1.000000
WR34	14.000000	INFINITY	7.000000
WR35	12.000000	INFINITY	3.000000
WR36	12.000000	1.000000	21.000000
WR37	6.000000	INFINITY	15.000000

RIGHTHAND SIDE RANGES

ROW	CURRENT RHS	ALLOWABLE INCREASE	ALLOWABLE DECREASE
2	150.000000	200.000000	0.000000
3	450.000000	INFINITY	0.000000
4	250.000000	250.000000	0.000000
5	150.000000	100.000000	0.000000
6	0.000000	0.000000	200.000000
7	0.000000	0.000000	250.000000
8	0.000000	0.000000	100.000000
9	100.000000	0.000000	100.000000
10	150.000000	0.000000	150.000000
11	100.000000	0.000000	100.000000
12	200.000000	0.000000	200.000000
13	200.000000	0.000000	200.000000
14	150.000000	0.000000	100.000000
15	100.000000	0.000000	100.000000
16	0.000000	150.000000	INFINITY
17	0.000000	0.000000	INFINITY
18	0.000000	200.000000	INFINITY
19	0.000000	250.000000	INFINITY
20	0.000000	0.000000	INFINITY
21	0.000000	150.000000	INFINITY
22	0.000000	100.000000	INFINITY
23	0.000000	0.000000	INFINITY
24	0.000000	150.000000	INFINITY
25	0.000000	100.000000	INFINITY
26	0.000000	150.000000	INFINITY
27	0.000000	100.000000	INFINITY
28	0.000000	0.000000	INFINITY
29	0.000000	0.000000	INFINITY
30	0.000000	200.000000	INFINITY
31	0.000000	200.000000	INFINITY
32	0.000000	0.000000	INFINITY
33	0.000000	0.000000	INFINITY
34	0.000000	0.000000	INFINITY
35	0.000000	150.000000	INFINITY
36	0.000000	100.000000	INFINITY

Summarized optimal solution and shipping costs (X indicates there is no route available):

Plant to warehouse

	w1	w2	w3
p1	150	0	X
p2	200	250	X
p3	0	150	100
p4	X	0	150

Warehouse to retailer

	r1	r2	r3	r4	r5	r6	r7
w1	100	150	100	0	X	X	X
w2	X	X	0	200	200	0	X
w3	X	X	X	0	0	150	100

Minimum cost: $17,100

Part B - Route Elimination (w2)

Initial thoughts:

Plants p1 and p2 produce 150+450=600 units and can only ship to warehouse w1. Warehouse w1 can only ship to retailers r1, r2, r3, and r4. These retailers have demand of 100+150+100+200=550 units. Supply demands demand by 50. These 50 units will not be sold since they cannot be shipped to a retailer that has unmet demand from consumers. They will not be shipped to a warehouse since it is cheaper for the plant to store the excess than to pay to have the extra units shipped to a warehouse.

Warehouse w3 can ship to retailers r4, r5, r6, and r7. These retailers have a total demand of 200+200+150+100=650 units. If retailer r4 is ignored demand is 450 units. Plants p3 and p4 are the only plants that can ship to warehouse w3; their total supply is 250+150=400. If retailer r3 is ignored, supply is 50 short of meeting demand. If the demand at retailer r3 is included, then supply is 250 units short. Most likely an optimal solution does not exist - the elimination of warehouse w2 leaves 50 units unsold and 50 customers without a unit to purchase.

Optimal solution output from LINDO:

We will need to remove any plant to warehouse link (pw) and warehouse to retailer link (wr) that has warehouse w2 as a connection. Specifically we remove pw12, pw22, pw32, pw42, wr23, wr24, wr25 and wr26.

```
     OBJECTIVE FUNCTION VALUE

        1)      17650.00

  VARIABLE        VALUE          REDUCED COST
      PW11       150.000000          0.000000
      PW21       400.000000          0.000000
      PW31         0.000000          1.000000
      PW33       250.000000          0.000000
```

```
      PW43       150.000000          0.000000
      WR11       100.000000          0.000000
      WR12       150.000000          0.000000
      WR13       100.000000          0.000000
      WR14       200.000000          0.000000
      WR34         0.000000          1.000000
      WR35       150.000000          0.000000
      WR36       150.000000          0.000000
      WR37       100.000000          0.000000


       ROW   SLACK OR SURPLUS     DUAL PRICES
        2)         0.000000          0.000000
        3)        50.000000          0.000000
        4)         0.000000          1.000000
        5)         0.000000          1.000000
        6)         0.000000          0.000000
        7)         0.000000         -1.000000
        8)         0.000000          0.000000
        9)         0.000000          0.000000
       10)         0.000000          0.000000
       11)         0.000000          0.000000
       12)       -50.000000         -1.000000
       13)         0.000000         -1.000000
       14)         0.000000         -1.000000
       15)       150.000000          0.000000
       16)       400.000000          0.000000
       17)         0.000000          0.000000
       18)       250.000000          0.000000
       19)       150.000000          0.000000
       20)       100.000000          0.000000
       21)       150.000000          0.000000
       22)       100.000000          0.000000
       23)       200.000000          0.000000
       24)         0.000000          0.000000
       25)       150.000000          0.000000
       26)       150.000000          0.000000
       27)       100.000000          0.000000

 NO. ITERATIONS=      11
```

Feasibility to ship all units:

Row 12 shows that there is a surplus of 50 units. Also of note is that the total cost increases from \$17,100 to \$17,650 to sell 50 less units.

Part C - Limited Warehouse shipping Capacity (w2)

In this scenario we are presented with the same situation as seen in problem 1 part A, except here warehouse w2 can only ship 100 units. This can be represented in LINDO as a constraint in the form of wr23 + wr24 + wr25 + wr26 < 100. We simply add this and rerun LINDO to find an optimal solution.

The total price to move the units from plant to warehouse to retailers costs a total of \$18,300, an increase of nearly 7%.

New LINDO output:

```
        LP OPTIMUM FOUND AT STEP     15
               OBJECTIVE FUNCTION VALUE
               1)      18300.00

         VARIABLE        VALUE          REDUCED COST
             PW11       150.000000          0.000000
             PW12         0.000000          8.000000
             PW21       350.000000          0.000000
             PW22       100.000000          0.000000
             PW31         0.000000          4.000000
             PW32         0.000000          2.000000
             PW33       250.000000          0.000000
             PW42         0.000000          9.000000
             PW43       150.000000          0.000000
             WR11       100.000000          0.000000
             WR12       150.000000          0.000000
             WR13       100.000000          0.000000
             WR14       150.000000          0.000000
             WR23         0.000000          7.000000
             WR24        50.000000          0.000000
             WR25        50.000000          0.000000
             WR26         0.000000          4.000000
             WR34         0.000000          4.000000
             WR35       150.000000          0.000000
             WR36       150.000000          0.000000
             WR37       100.000000          0.000000

              ROW   SLACK OR SURPLUS     DUAL PRICES
               2)         0.000000          5.000000
               3)         0.000000          1.000000
               4)         0.000000          0.000000
               5)         0.000000          2.000000
               6)         0.000000          3.000000
               7)         0.000000        -11.000000
               8)         0.000000         -8.000000
               9)         0.000000        -11.000000
              10)         0.000000        -16.000000
              11)         0.000000        -17.000000
              12)         0.000000        -18.000000
              13)         0.000000        -21.000000
              14)         0.000000        -23.000000
              15)         0.000000        -23.000000
              16)         0.000000        -17.000000
              17)       150.000000          0.000000
              18)         0.000000          0.000000
              19)       350.000000          0.000000
              20)       100.000000          0.000000
              21)         0.000000          0.000000
              22)         0.000000          0.000000
              23)       250.000000          0.000000
              24)         0.000000          0.000000
              25)       150.000000          0.000000
              26)       100.000000          0.000000
              27)       150.000000          0.000000
              28)       100.000000          0.000000
              29)       150.000000          0.000000
              30)         0.000000          0.000000
              31)        50.000000          0.000000
              32)        50.000000          0.000000
              33)         0.000000          0.000000
              34)         0.000000          0.000000
              35)       150.000000          0.000000
              36)       150.000000          0.000000
              37)       100.000000          0.000000

        NO. ITERATIONS=      15
```

Updated plant to warehouse table:

	w1	w2	w3
p1	150	0	X
p2	350	100	X
p3	0	0	250
p4	X	0	150

Updated warehouse to retailer table:

	r1	r2	r3	r4	r5	r6	r7
w1	100	150	100	150	X	X	X
w2	X	X	0	50	50	0	X
w3	X	X	X	0	150	150	100

Part D

Function to Minimize

$$\text{minimize}\left(\sum_{i=1}^{n}\sum_{j=1}^{q} cp(i,j) * p_i w_j + \sum_{j=1}^{q}\sum_{k=1}^{m} cw(j,k) * w_j r_k\right)$$

Supply Constraint

$$\sum_{j=1}^{q} p_i w_j \leq s_i \quad for\ i = 1 \ldots n$$

Receiving and Shipping Constraint

$$\sum_{i=1}^{q} p_i w_j \geq \sum_{k=1}^{m} w_j r_k \quad for\ j = 1 \ldots q$$

Demand Constraint

$$\sum_{j=1}^{q} w_j r_k \geq d_k \quad for\ k = 1 \ldots m$$

Non-negative Constraint

$$cp(i,j) \geq 0\ and\ cw(j,k) \geq 0 \quad for\ all\ edges$$
$$n > 0 \quad q > 0 \quad m > 0$$
$$s_i > 0\ and\ d_k > 0$$

Paths Constraint

$$\exists\left|E_{i \to j}\right| \quad for\ i = 1 \ldots n\ and\ j = 1 \ldots q$$

$$\exists\left|E_{j \to k}\right| \quad for\ j = 1 \ldots q\ and\ k = 1 \ldots m$$

Problem 2: Mixture Problem

Each salad must contain:

- At least 15 grams of protein
- At least 2 and at most 8 grams of fat
- At least 4 grams of carbohydrates
- At most 200 milligrams of sodium
- At least 40% leafy greens by mass.

The nutritional contents of these ingredients (per 100 grams) and cost are

Ingredient	Energy (kcal)	Protein (grams)	Fat (grams)	Carbohydrate (grams)	Sodium (mg)	Cost (100g)
Tomato	21	0.85	0.33	4.64	9.00	$1.00
Lettuce	16	1.62	0.20	2.37	28.00	$0.75
Spinach	40	2.86	0.39	3.63	65.00	$0.50
Carrot	41	0.93	0.24	9.58	69.00	$0.50
Sunflower Seeds	585	23.4	48.7	15.00	3.80	$0.45
Smoked Tofu	120	16.00	5.00	3.00	120.00	$2.15
Chickpeas	164	9.00	2.6	27.0	78.00	$0.95
Oil	884	0	100.00	0	0	$2.00

Part A

i. For notation, we use the first three letters of each ingredient to clarify which ingredient is which.

Objective(Calories):

MIN (21 x tom) + (16 x let) + (40 x spi) + (41 x car) + (585 x sun) + (120 x smo) + (164 x chi) + (884 x oil)

Constraints

Protein:

(.85 x tom) + (1.62 x let) + (2.86 x spi) + (.93 x car) + (23.4 x sun) + (16 x smo) + (9 x chi) + (0 x oil) >= 15

Fat:

(.33 x tom) + (.20 x let) + (.39 x spi) + (.24 x car) + (48.7 x sun) + (5 x smo) + (2.6 x chi) + (100 x oil) >= 2
(.33 x tom) + (.20 x let) + (.39 x spi) + (.24 x car) + (48.7 x sun) + (5 x smo) + (2.6 x chi) + (100 x oil) <= 8

Carbohydrates:

(4.64 x tom) + (2.37 x let) + (3.63 x spi) + (9.58 x car) + (15 x sun) + (3 x smo) + (27 x chi) + (0 x oil) >= 4

Sodium:

(9 x tom) + (28 x let) + (65 x spi) + (69 x car) + (3.8 x sun) + (120 x smo) + (78 x chi) + (0 x oil) <= 200

Leafy Greens by Mass:

(let + spi)/(tom + let + spi + car + sun + smo + chi + oil) = .4

Non negative:

tom, let, spi, car, sun, smo, chi, oil >= 0

ii. For this problem we once again used lindo to find our solution. Using the equations we created above to develop the program below:

```
min     ETOT

ST
        PTOT >= 15
        FTOT >= 2
        FTOT <= 8
        CTOT >= 4
        STOT <= 200

        0.6 LET + 0.6 SPI - 0.4 TOM - 0.4 CAR - 0.4 SUN - 0.4 TOF - 0.4 CHI - 0.4 OIL >= 0

        TOM >= 0
        SPI >= 0
        LET >= 0
        CAR >= 0
        SUN >= 0
        TOF >= 0
        CHI >= 0
        OIL >= 0

        ETOT - 21 TOM - 16 LET - 40 SPI - 41 CAR - 585 SUN - 120 TOF - 164 CHI - 884 OIL = 0
        PTOT - 0.85 TOM - 1.62 LET - 2.86 SPI - 0.93 CAR - 23.4 SUN - 16 TOF - 9 CHI - 0 OIL = 0
        FTOT - 0.33 TOM - 0.20 LET - 0.39 SPI - 0.24 CAR - 48.7 SUN - 5 TOF - 2.6 CHI - 100 OIL = 0
        CTOT - 4.64 TOM - 2.37 LET - 3.63 SPI - 9.58 CAR - 15 SUN - 3 TOF - 27 CHI - 0 OIL = 0
        STOT - 9 TOM - 28 LET - 65 SPI -  69 CAR - 3.8 SUN - 120 TOF - 78 CHI - 0 OIL = 0
        DTOT - 1 TOM - 0.75 LET - 0.50 SPI - 0.5 CAR - 0.45 SUN - 2.15 TOF - 0.95 CHI - 2.00 OIL =
0
END
```

Using the solution of .575 lettuce and .8973 smoked tofu we confirmed each constraint:

- At least 15 grams of protein
  (1.62 x .575 lettuce)+ (16 x.8793 smoked tofu) = 14.99
- At least 2 and at most 8 grams of fat
  (.2 x .575 lettuce)+ (5 x.8793 smoked tofu) = 4.5
- At least 4 grams of carbohydrates
  (2.37 x .575 lettuce)+ (3 x.8793 smoked tofu) = 4
- At most 200 milligrams of sodium
  (.2 x .575 lettuce)+ (5 x.8793 smoked tofu) = 121.6
- At least 40% leafy greens by mass.
  .575/(.575 + .8793) = .395

Total calories for low calorie options is: **114.7**

iii. What is the cost of the low calorie salad?

Cost = (.75 x .575 lettuce)+ (2.15 x.8793 smoked tofu) = **$2.32**

Part B

Veronica realizes that it is also important to minimize the cost associated with the new salad. Unfortunately some of the ingredients can be expensive. Determine the combination of ingredients that minimizes cost.

i. Formulate the problem as a linear program with an objective function and all constraints.

   For this problem, we use the same constraints from part A but adjust for the objective:

   Objective(cost):
   MIN (1.00 x tom) + (.75 x let) + (.50 x spi) + (.50 x car) + (.45 x sun) + (2.15 x smo) + (.95 x chi) + (2.00 x oil)

ii. Determine the optimal solution for the linear program using any software you want. Include a copy of the code/file in the report.

```
        min     DTOT

ST
        PTOT >= 15
        FTOT >= 2
        FTOT <= 8
        CTOT >= 4
        STOT <= 200

        0.6 LET + 0.6 SPI - 0.4 TOM - 0.4 CAR - 0.4 SUN - 0.4 TOF - 0.4 CHI - 0.4 OIL >= 0


        TOM >= 0
        SPI >= 0
        LET >= 0
        CAR >= 0
        SUN >= 0
        TOF >= 0
        CHI >= 0
        OIL >= 0

        ETOT - 21 TOM - 16 LET - 40 SPI - 41 CAR - 585 SUN - 120 TOF - 164 CHI - 884 OIL = 0
        PTOT - 0.85 TOM - 1.62 LET - 2.86 SPI - 0.93 CAR - 23.4 SUN - 16 TOF - 9 CHI - 0 OIL = 0
        FTOT - 0.33 TOM - 0.20 LET - 0.39 SPI - 0.24 CAR - 48.7 SUN - 5 TOF - 2.6 CHI - 100 OIL = 0
        CTOT - 4.64 TOM - 2.37 LET - 3.63 SPI - 9.58 CAR - 15 SUN - 3 TOF - 27 CHI - 0 OIL = 0
        STOT - 9 TOM - 28 LET - 65 SPI -  69 CAR - 3.8 SUN - 120 TOF - 78 CHI - 0 OIL = 0
        DTOT - 1 TOM - 0.75 LET - 0.50 SPI - 0.5 CAR - 0.45 SUN - 2.15 TOF - 0.95 CHI - 2.00 OIL = 0
END
```

LINDO gave us the resulting price of **DTOT = $1.55**

iii. How many calories are in the low cost salad?

   LINDO gave us a result of ETOT = **278.488 calories**

Part C

Compare the results from part A and B. Veronica's goal is to create a Very Veggie Salad that is both low calorie and low cost. She would like to sell the salad for $5.00 and still have a profit of at least $3.00. However if she can advertise that the salad has under 250 calories then she may be able to sell more.

i. Suggest some possible ways that she select a combination of ingredients that is "near optimal" for both objectives. This is a type of multi-objective optimization.

   -Veronica wants to sell at $5.00 and make profit of $3.00, thus the target cost is <= $2.00
   -We also want to keep calories (ETOT) under 250.
   One way would be to set a condition that ETOT <= 250, and then minimize DTOT. This results in: DTOT = $1.62 and ETOT = 250.
   Another way would be to set a condition that DTOT <= 2.00, and minimize ETOT. This results in: DTOT = $2.00 and ETOT = 134.76.

ii. What combination of ingredient would you suggest and what is the associated cost and calorie.

   We would recommend the first solution (set ETOT max = 250 and minimize cost). The resulting values are .55 Spinach, .03 sunflower seeds, .77 tofu, and 0 for all other ingredients.

iii. Note: There is not one "right" answer. Discuss how you derived your solution.

   We made this choice due to assuming that reducing calories below 250 will not impact sales (as the prompt seems to indicate). So, sales are maximized, while also maximizing the amount of profit made from each sale (since we minimized cost). Using this strategy works well because low calories and low cost are also directly correlated.

Problem 3: Solving shortest path problems using linear programming

We used Lindo to solve for the shortest paths using this code:

```
max db

ST
   da = 0
   db - da <= 2
   dc - da <= 3
   dd - da <= 8
   dh - da <= 9
   da - db <= 4
   dc - db <= 5
   de - db <= 7
   df - db <= 4
   dd - dc <= 10
   db - dc <= 5
   dg - dc <= 9
   di - dc <= 11
   df - dc <= 4
   da - dd <= 8
   dg - dd <= 2
```

```
   dj - dd <= 5
   df - dd <= 1
   dh - de <= 5
   dc - de <= 4
   di - de <= 10
   di - df <= 2
   dg - df <= 2
   dd - dg <= 2
   dj - dg <= 8
   dk - dg <= 12
   di - dh <= 5
   dk - dh <= 10
   da - di <= 20
   dk - di <= 6
   dj - di <= 2
   dm - di <= 12
   di - dj <= 2
   dk - dj <= 4
   dl - dj <= 5
   dh - dk <= 10
   dm - dk <= 10
   dm - dl <= 2

END
```

Part A

Target	Shortest Distance From a
b	2
c	3
d	8
e	9
f	6
g	8
h	9
i	8
j	10
k	14
l	15
m	17

Part B

We get an unbounded error (52) and a large result (0.9999990E+08) when we try to find the path from a to z.

Part C

Origin	Distance to m
a	17
b	15
c	15
d	12
e	19
f	11
g	14
h	14
i	9
j	7
k	10
l	2

Since “m” is the last vertex in the directed graph, there is a relatively simple way to solve the problem of finding the distances from all the other vertices to m. Namely, we reverse the direction of all the edges in the graph, and then solve for the Max. The constraints would therefore be reversed. For example, db - da <= 2 would be changed to da - db <= 2.

Part D

You can calculate the shortest path from vertex x to vertex i, then calculate the shortest path from vertex i to vertex y. Then add the paths from the two tables together (e.g. to get from a to h through i, you do 8 + 16 = 24).

Origin	Distance to i
a	8
b	6
c	6
d	3
e	10
f	2
g	5

h	5
j	2
k	15
l	UNBOUNDED
m	UNBOUNDED

Target	Distance From i
a	20
b	22
c	23
d	28
e	29
f	26
g	28
h	16
j	2
k	6
l	7
m	9

Rows are end points, columns are starting points. No column for l or m as there is no path to i from those nodes.

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
a	0	26	26	23	30	22	25	25	20	22	35
b	30	0	28	25	32	24	27	27	22	24	37
c	31	29	0	26	33	25	28	28	23	25	38
d	36	34	34	0	38	30	33	33	28	30	43
e	37	35	35	32	0	31	34	34	29	31	44
f	34	32	32	29	36	0	31	31	26	28	41
g	36	34	34	31	38	30	0	33	28	30	43

h	24	22	22	19	26	18	21	0	16	18	31
i	8	6	6	3	10	2	5	5	0	2	15
j	10	8	8	5	12	4	7	7	2	0	17
k	14	12	12	9	16	8	11	11	6	8	0
l	15	13	13	10	17	9	12	12	7	9	22
m	17	15	15	12	19	11	14	14	9	11	24