

Homework 1 - Truth Tables & Linear Programming with AMPL

Daniel Carpenter and Effouehi Moody

January 2022

Contents

1	Problem 1	3
2	Problem 2	5
3	Problem 3	6
3.1	Task a	6
4	Problem 4	9
4.1	Task a	9
4.2	Task b	13
4.3	Task c	16
4.4	Task d	19
5	Problem 5	22
5.1	Base Mathematical Formulation and Code	22
5.2	Task a	24
5.3	Task b	26
5.4	Task c	28
5.5	Task d	30
5.6	Task e	33

6	Problem 6	34
6.1	Task a - c	34
6.2	Task d:	37
6.3	Task e:	37

1 Problem 1

Efflower Moody
Daniel Carpenter

Homework 1

Question 1

Let make a truth table with T for Truth and F for False

Cases	Gregon	Tyxlin	CateLyn	Tyxlin: only one of us is a truth-teller	CateLyn: At least one of us is a truth-teller	
1	T	T	T	F	T	x
2	T	T	F	F	T	x
3	T	F	T	F	T	
4	T	F	F	T	T	x
5	F	T	T	F	T	x
6	F	T	F	T	T	x
7	F	F	T	T	T	x
8	F	F	F	F	F	x

-Tyxlin and CateLyn cannot be telling the truth at the same time. They also cannot lie at the same time. If one is lying, the other is telling the truth so case 1, 4, 5 and 8 are impossible.

-Case 2: Gregon and Tyxlin are truth-tellers. If Gregon is a truth-teller, that means only 1 of them is a truth-teller which contradicts that both Tyxlin and Gregon are truth-tellers. So case 2 is impossible.

-Case 6: Gregon is liar. This means not exactly one person is truth-teller but CateLyn is a liar and Tyxlin remains the only truth-teller. This case is impossible.

Question 1 (Continued)

Case 7: Greger is a liar. This means not exactly 1 person is truth-teller but Tyxin is a liar and Catelyn remain the only truth-teller. This case is also impossible.

Case 3: Tyxin is a liar, that means Greger didn't say "one of us is truth-teller", he probably said "2 of us are truth-tellers", then Greger and Catelyn are truth-tellers and Tyxin is a liar. This case is correct.

Therefore we can say that Greger and Catelyn are truth-tellers and Tyxin is a liar.

2 Problem 2

Question 2

Let G be gump and P be Rulok.

Yes	No	Does gump mean yes?	
		Yoda is a liar	Yoda is a truth-teller
P	G	P	G
G	P	P	G

Case 1: Yoda is a truth-teller

* If gump means Yes, he should answer gump.

* If gump means No, he should answer gump.

Case 2: Yoda is a liar

* If gump means Yes, then Rulok means No and he should answer Rulok.

* If gump means No, then Rulok means Yes and he should answer Rulok.

Therefore, Yoda is a liar

3 Problem 3

3.1 Task a

3.1.1 Decision Variables

bondA: dollars $\in \mathbb{R}$ to invest in bond A

bondB: dollars $\in \mathbb{R}$ to invest in bond B

bondC: dollars $\in \mathbb{R}$ to invest in bond C

bondD: dollars $\in \mathbb{R}$ to invest in bond D

bondE: dollars $\in \mathbb{R}$ to invest in bond E

3.1.2 Objective Function

- Maximize the Expected Earnings of the portfolio

$$\text{Maximize } Z = (0.043 \times \text{bondA}) + (0.027 \times \text{bondB}) + (0.025 \times \text{bondC}) + (0.022 \times \text{bondD}) + (0.045 \times \text{bondE})$$

3.1.3 Constraints

C1: Budget to invest is \$10 MM or less

$$\text{budget} : \text{bondA} + \text{bondB} + \text{bondC} + \text{bondD} + \text{bondE} \leq 10$$

C2: At least \$4 million must be invested in government and agency bonds

$$\text{govtAndAgency} : \text{bondB} + \text{bondC} + \text{bondD} \geq 4$$

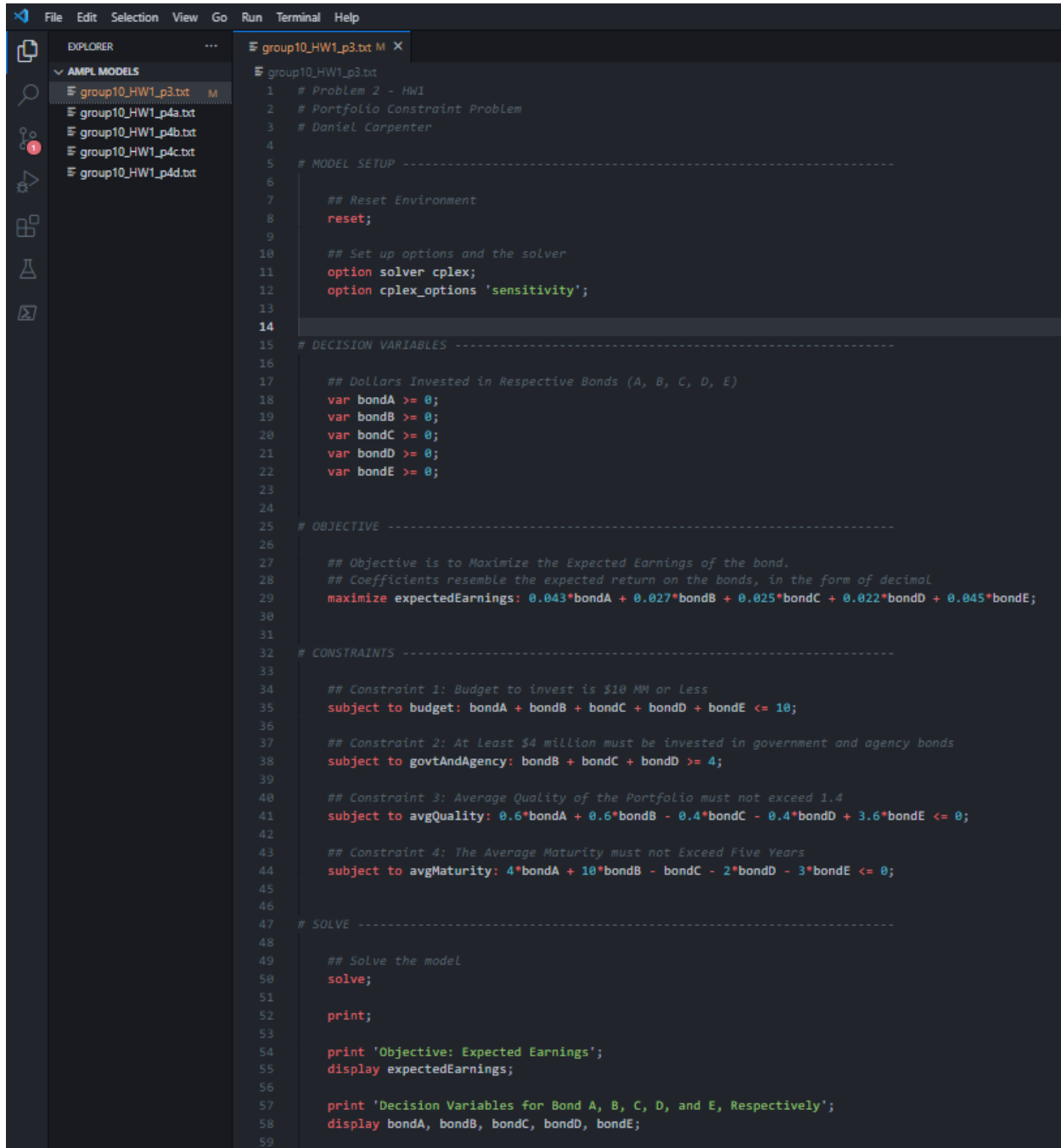
C3: Average Quality of the Portfolio must not exceed 1.4

$$\text{avgQuality} : (0.6 \times \text{bondA}) + (0.6 \times \text{bondB}) - (0.4 \times \text{bondC}) - (0.4 \times \text{bondD}) + (3.6 \times \text{bondE}) \leq 0$$

C4: The Average Maturity must not Exceed Five Years

$$\text{avgMaturity} : (4 \times \text{bondA}) + (10 \times \text{bondB}) - (1 \times \text{bondC}) - (2 \times \text{bondD}) - (3 \times \text{bondE}) \leq 0$$

3.1.4 Code



```
1 # Problem 2 - HW1
2 # Portfolio Constraint Problem
3 # Daniel Carpenter
4
5 # MODEL SETUP -----
6
7 ## Reset Environment
8 reset;
9
10 ## Set up options and the solver
11 option solver cplex;
12 option cplex_options 'sensitivity';
13
14
15 # DECISION VARIABLES -----
16
17 ## Dollars Invested in Respective Bonds (A, B, C, D, E)
18 var bondA >= 0;
19 var bondB >= 0;
20 var bondC >= 0;
21 var bondD >= 0;
22 var bondE >= 0;
23
24
25 # OBJECTIVE -----
26
27 ## Objective is to Maximize the Expected Earnings of the bond.
28 ## Coefficients resemble the expected return on the bonds, in the form of decimal
29 maximize expectedEarnings: 0.043*bondA + 0.027*bondB + 0.025*bondC + 0.022*bondD + 0.045*bondE;
30
31
32 # CONSTRAINTS -----
33
34 ## Constraint 1: Budget to invest is $10 MM or Less
35 subject to budget: bondA + bondB + bondC + bondD + bondE <= 10;
36
37 ## Constraint 2: At Least $4 million must be invested in government and agency bonds
38 subject to govtAndAgency: bondB + bondC + bondD >= 4;
39
40 ## Constraint 3: Average Quality of the Portfolio must not exceed 1.4
41 subject to avgQuality: 0.6*bondA + 0.6*bondB - 0.4*bondC - 0.4*bondD + 3.6*bondE <= 0;
42
43 ## Constraint 4: The Average Maturity must not Exceed Five Years
44 subject to avgMaturity: 4*bondA + 10*bondB - bondC - 2*bondD - 3*bondE <= 0;
45
46
47 # SOLVE -----
48
49 ## Solve the model
50 solve;
51
52 print;
53
54 print 'Objective: Expected Earnings';
55 display expectedEarnings;
56
57 print 'Decision Variables for Bond A, B, C, D, and E, Respectively';
58 display bondA, bondB, bondC, bondD, bondE;
59
```

3.1.5 Output

```
ampl: model group10_HW1_p3.txt
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 0.2983636364
3 dual simplex iterations (1 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;

Objective: Expected Earnings
expectedEarnings = 0.298364

Decision Variables for Bond A, B, C, D, and E, Respectively
bondA = 2.18182
bondB = 0
bondC = 7.36364
bondD = 0
bondE = 0.454545
```


4 Problem 4

4.1 Task a

4.1.1 Decision Variables

tv = the number of minutes $\in \mathbb{R}$ to air advertising on the *television* medium

magazine = the number of pages $\in \mathbb{I}$ to to advertise on the *magazine* medium

4.1.2 Objective Function

- Maximize the total audience reach

$$\text{Maximize } Z = (1.8 \times tv) + (1.0 \times magazine)$$

4.1.3 Constraints

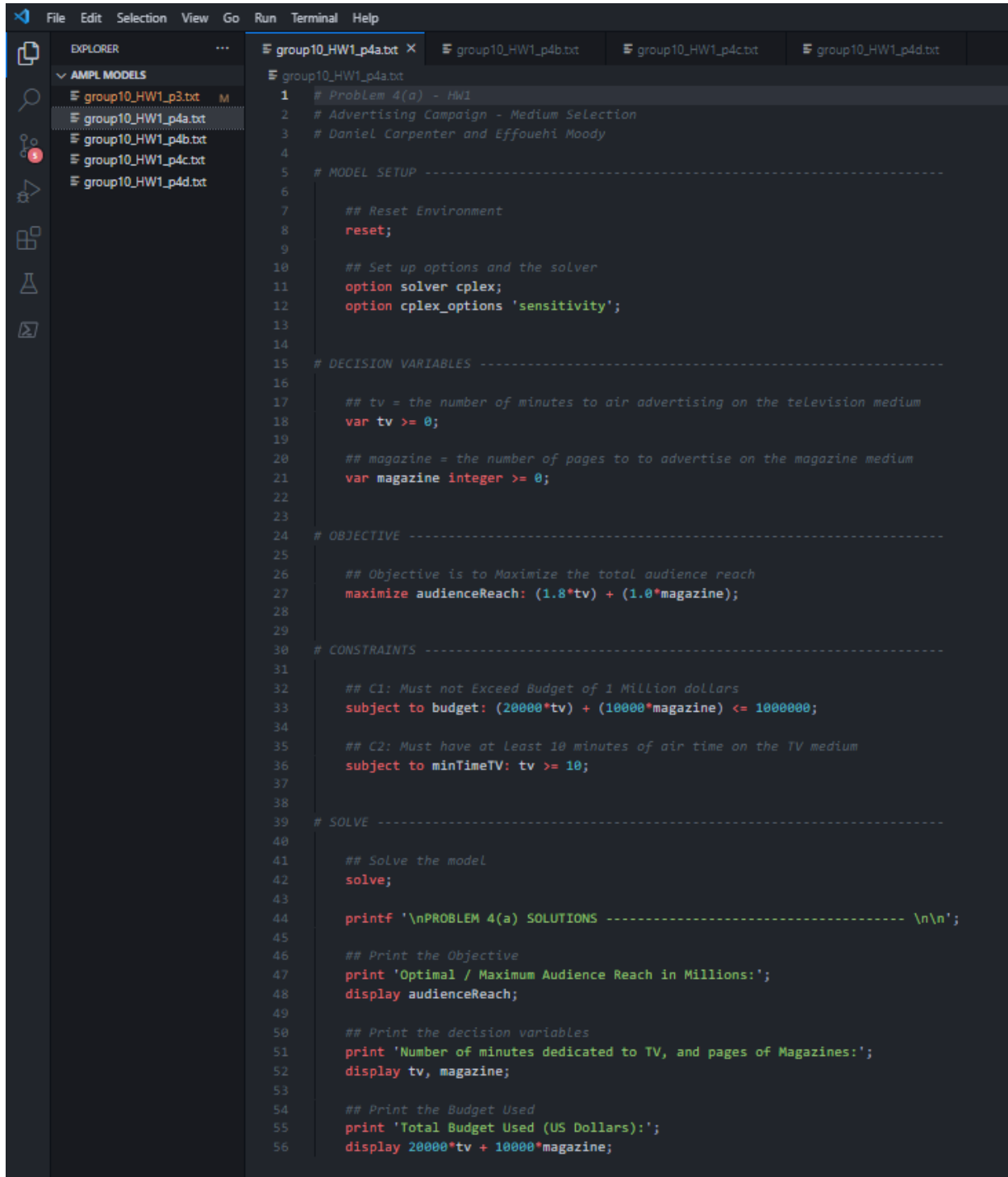
C1: Must not Exceed Budget of 1 Million dollars

$$\text{budget} : (20,000 \times tv) + (10,000 \times magazine) \leq 1,000,000$$

C2: Must have at least 10 minutes of air time on the TV medium

$$\text{minTimeTV} : tv \geq 10$$

4.1.4 Code



```
1  # Problem 4(a) - HW1
2  # Advertising Campaign - Medium Selection
3  # Daniel Carpenter and Effouehi Moody
4
5  # MODEL SETUP -----
6
7  ## Reset Environment
8  reset;
9
10 ## Set up options and the solver
11 option solver cplex;
12 option cplex_options 'sensitivity';
13
14
15 # DECISION VARIABLES -----
16
17 ## tv = the number of minutes to air advertising on the television medium
18 var tv >= 0;
19
20 ## magazine = the number of pages to to advertise on the magazine medium
21 var magazine integer >= 0;
22
23
24 # OBJECTIVE -----
25
26 ## Objective is to Maximize the total audience reach
27 maximize audienceReach: (1.8*tv) + (1.0*magazine);
28
29
30 # CONSTRAINTS -----
31
32 ## C1: Must not Exceed Budget of 1 Million dollars
33 subject to budget: (20000*tv) + (10000*magazine) <= 1000000;
34
35 ## C2: Must have at Least 10 minutes of air time on the TV medium
36 subject to minTimeTV: tv >= 10;
37
38
39 # SOLVE -----
40
41 ## Solve the model
42 solve;
43
44 printf '\nPROBLEM 4(a) SOLUTIONS ----- \n\n';
45
46 ## Print the Objective
47 print 'Optimal / Maximum Audience Reach in Millions: ';
48 display audienceReach;
49
50 ## Print the decision variables
51 print 'Number of minutes dedicated to TV, and pages of Magazines: ';
52 display tv, magazine;
53
54 ## Print the Budget Used
55 print 'Total Budget Used (US Dollars): ';
56 display 20000*tv + 10000*magazine;
```

4.1.5 Output

```
ampl: model group10_HW1_p4a.txt
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 98
0 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;

PROBLEM 4(a) SOLUTIONS -----

Optimal / Maximum Audience Reach in Millions:
audienceReach = 98

Number of minutes dedicated to TV, and pages of Magazines:
tv = 10
magazine = 80

Total Budget Used (US Dollars):
20000*tv + 10000*magazine = 1e+06
```

4.1.6 Solving Problem 4(a) Graphically by Hand

(1) $20,000 \cdot tv + 10,000(0) \leq 1,000,000$
 $tv \leq 50 \rightarrow (50, 0)$

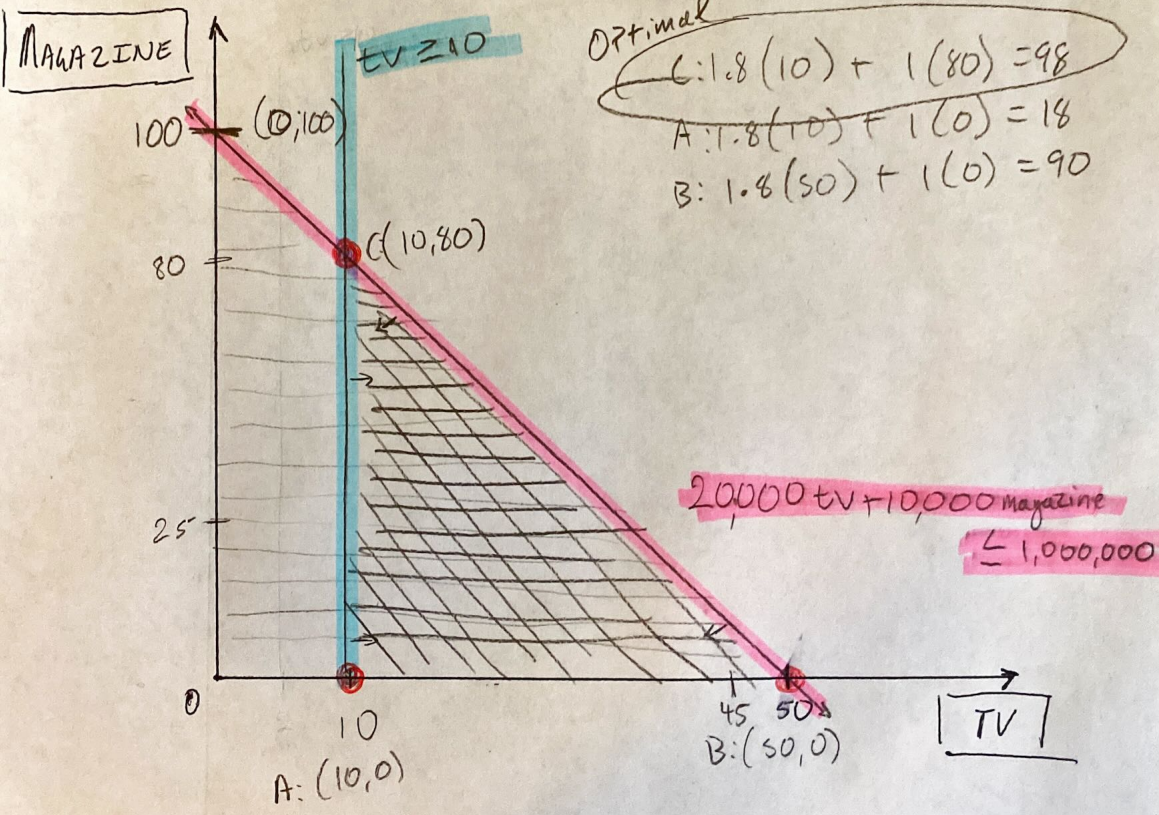
0 tv $10,000 \text{ Magazine} \leq 1,000,000$
 $\text{Magazine} \leq 100 \rightarrow (0, 100)$

(2)

$tv \geq 10$
 $\rightarrow (10, 0)$

Obj: $1.8 \text{ tv} + 1 \text{ magazine}$
 Gradient: $\left(\frac{df}{dx}, \frac{df}{dy}\right) = (1.8, 1)$

Optimal
 $C: 1.8(10) + 1(80) = 98$
 $A: 1.8(10) + 1(0) = 18$
 $B: 1.8(50) + 1(0) = 90$



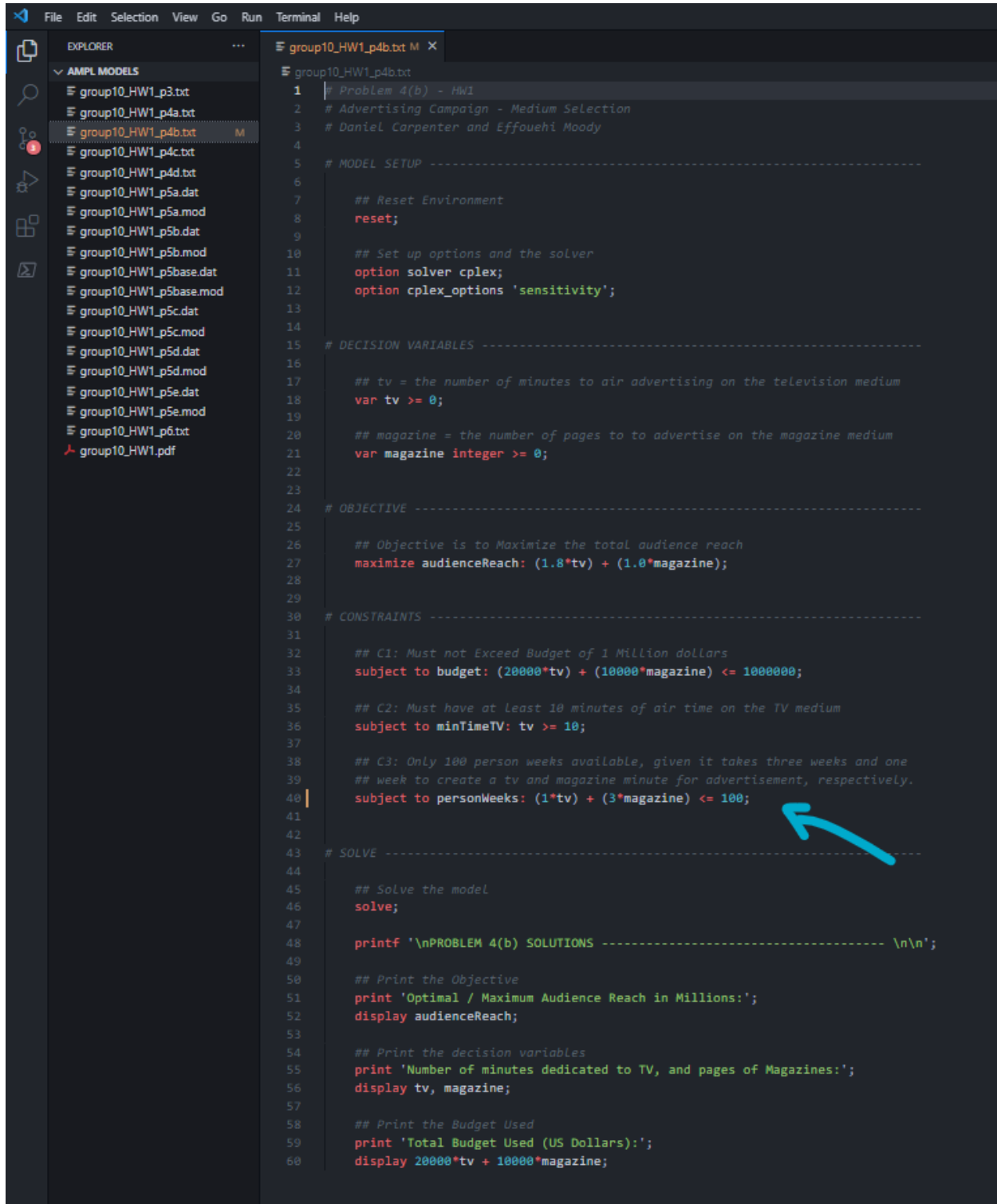
4.2 Task b

4.2.1 Additional Constraint: Labor Time

C3: Only 100 person weeks available, given it takes three weeks and one week to create a `magazine` page and `tv` minute for advertisement, respectively.

$$personWeeks : (1 \times tv) + (3 \times magazine) \leq 100$$

4.2.2 Code



```
1 # Problem 4(b) - HW1
2 # Advertising Campaign - Medium Selection
3 # Daniel Carpenter and Effouehi Moody
4
5 # MODEL SETUP -----
6
7 ## Reset Environment
8 reset;
9
10 ## Set up options and the solver
11 option solver cplex;
12 option cplex_options 'sensitivity';
13
14 # DECISION VARIABLES -----
15
16 ## tv = the number of minutes to air advertising on the television medium
17 var tv >= 0;
18
19 ## magazine = the number of pages to to advertise on the magazine medium
20 var magazine integer >= 0;
21
22
23 # OBJECTIVE -----
24
25 ## Objective is to Maximize the total audience reach
26 maximize audienceReach: (1.8*tv) + (1.0*magazine);
27
28
29 # CONSTRAINTS -----
30
31 ## C1: Must not Exceed Budget of 1 Million dollars
32 subject to budget: (20000*tv) + (10000*magazine) <= 1000000;
33
34 ## C2: Must have at Least 10 minutes of air time on the TV medium
35 subject to minTimeTV: tv >= 10;
36
37 ## C3: Only 100 person weeks available, given it takes three weeks and one
38 ## week to create a tv and magazine minute for advertisement, respectively.
39 subject to personWeeks: (1*tv) + (3*magazine) <= 100;
40
41
42 # SOLVE -----
43
44 ## Solve the model
45 solve;
46
47 printf '\nPROBLEM 4(b) SOLUTIONS ----- \n\n';
48
49 ## Print the Objective
50 print 'Optimal / Maximum Audience Reach in Millions:';
51 display audienceReach;
52
53 ## Print the decision variables
54 print 'Number of minutes dedicated to TV, and pages of Magazines:';
55 display tv, magazine;
56
57 ## Print the Budget Used
58 print 'Total Budget Used (US Dollars):';
59 display 20000*tv + 10000*magazine;
60
```

4.2.3 Output

```
ampl: model group10_HW1_p4b.txt
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal integer solution; objective 92
2 MIP simplex iterations
0 branch-and-bound nodes

suffix up OUT;
suffix down OUT;
suffix current OUT;

PROBLEM 4(b) SOLUTIONS -----

Optimal / Maximum Audience Reach in Millions:
audienceReach = 92

Number of minutes dedicated to TV, and pages of Magazines:
tv = 40
magazine = 20

Total Budget Used (US Dollars):
20000*tv + 10000*magazine = 1e+06
```

4.3 Task c

4.3.1 Additional Constraint: Radio Advertising Medium

4.3.2 Decision Variables

tv = the number of minutes $\in \mathbb{R}$ to air advertising on the *television* medium

magazine = the number of pages $\in \mathbb{I}$ to to advertise on the *magazine* medium **radio** = the number of minutes $\in \mathbb{R}$ to air advertising on the *radio* medium

4.3.3 Objective Function

- Maximize the total audience reach

$$\text{Maximize } Z = (1.80 \times tv) + (1.00 \times magazine) + (0.25 \times radio)$$

4.3.4 New Constraints

C1: Must not Exceed Budget of 1 Million dollars

$$\text{budget} : (20,000 \times tv) + (10,000 \times magazine) + (2,000 \times radio) \leq 1,000,000$$

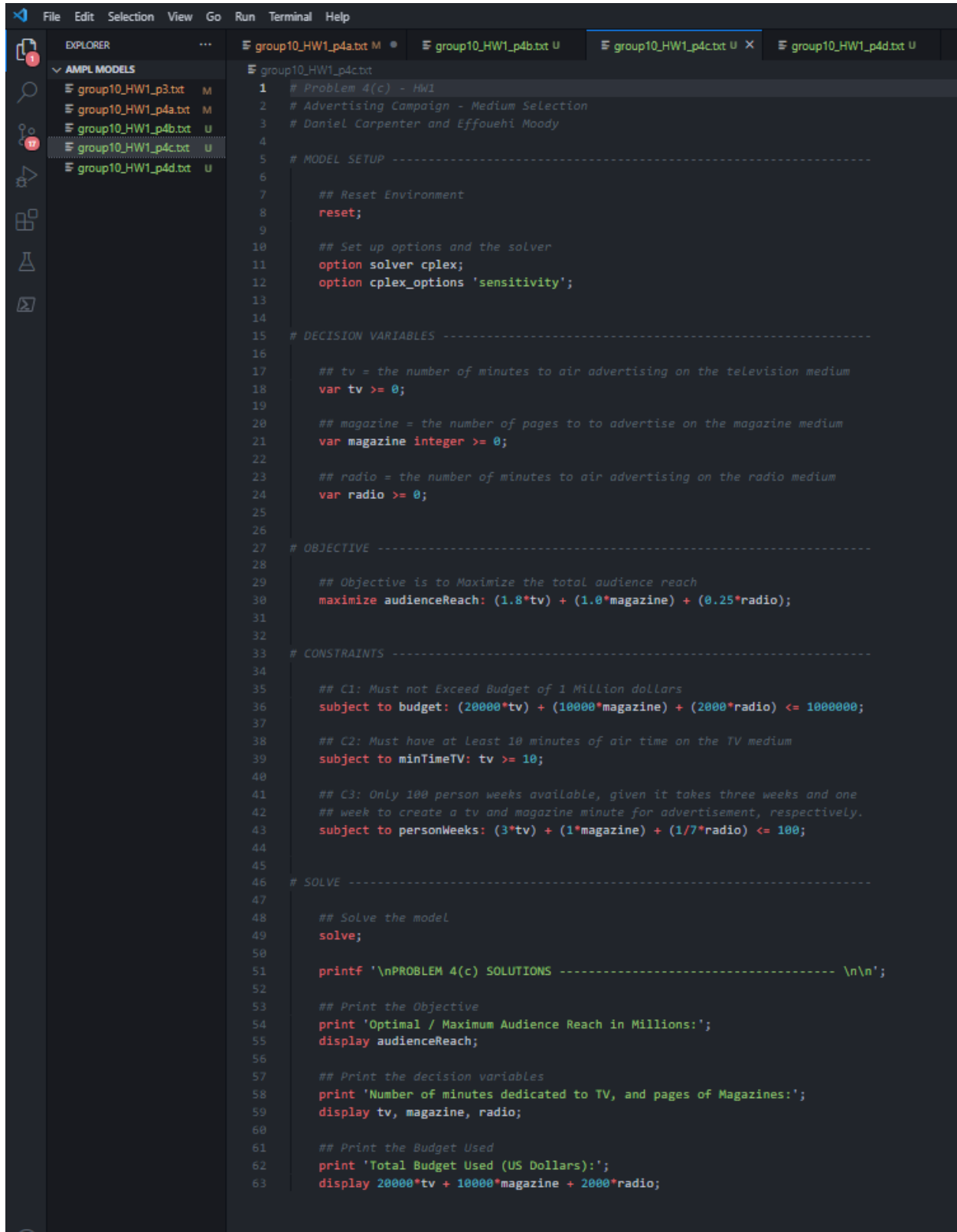
C2: Must have at least 10 minutes of air time on the TV medium

$$\text{minTimeTV} : tv \geq 10$$

C3: Only 100 person weeks available, given it takes three weeks and one week to create a **tv** and **magazine** minute for advertisement, respectively. It only takes one day for **radio**.

$$\text{personWeeks} : (3 \times tv) + (1 \times magazine) + \left(\frac{1}{7} \times radio\right) \leq 100$$

4.3.5 Code



```
1 # Problem 4(c) - HW1
2 # Advertising Campaign - Medium Selection
3 # Daniel Carpenter and Effouehi Moody
4
5 # MODEL SETUP -----
6
7 ## Reset Environment
8 reset;
9
10 ## Set up options and the solver
11 option solver cplex;
12 option cplex_options 'sensitivity';
13
14
15 # DECISION VARIABLES -----
16
17 ## tv = the number of minutes to air advertising on the television medium
18 var tv >= 0;
19
20 ## magazine = the number of pages to to advertise on the magazine medium
21 var magazine integer >= 0;
22
23 ## radio = the number of minutes to air advertising on the radio medium
24 var radio >= 0;
25
26
27 # OBJECTIVE -----
28
29 ## Objective is to Maximize the total audience reach
30 maximize audienceReach: (1.8*tv) + (1.0*magazine) + (0.25*radio);
31
32
33 # CONSTRAINTS -----
34
35 ## C1: Must not Exceed Budget of 1 Million dollars
36 subject to budget: (20000*tv) + (10000*magazine) + (2000*radio) <= 1000000;
37
38 ## C2: Must have at Least 10 minutes of air time on the TV medium
39 subject to minTimeTV: tv >= 10;
40
41 ## C3: Only 100 person weeks available, given it takes three weeks and one
42 ## week to create a tv and magazine minute for advertisement, respectively.
43 subject to personWeeks: (3*tv) + (1*magazine) + (1/7*radio) <= 100;
44
45
46 # SOLVE -----
47
48 ## Solve the model
49 solve;
50
51 printf '\nPROBLEM 4(c) SOLUTIONS ----- \n\n';
52
53 ## Print the Objective
54 print 'Optimal / Maximum Audience Reach in Millions:';
55 display audienceReach;
56
57 ## Print the decision variables
58 print 'Number of minutes dedicated to TV, and pages of Magazines:';
59 display tv, magazine, radio;
60
61 ## Print the Budget Used
62 print 'Total Budget Used (US Dollars):';
63 display 20000*tv + 10000*magazine + 2000*radio;
```

4.3.6 Output

```
ampl: model group10_HW1_p4c.txt
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 118
1 dual simplex iterations (1 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;

PROBLEM 4(c) SOLUTIONS -----

Optimal / Maximum Audience Reach in Millions:
audienceReach = 118

Number of minutes dedicated to TV, and pages of Magazines:
tv = 10
magazine = 0
radio = 400

Total Budget Used (US Dollars):
20000*tv + 10000*magazine + 2000*radio = 1e+06
```

4.4 Task d

4.4.1 Additional Constraints: Minimum Magazine and Maximum Radio Requirements

C4: Must sign up for at least 2 magazine pages

$$\text{minMagazines} : \text{magazine} \geq 2$$

C5: Must to exceed 120 minutes of radio

$$\text{maxRadio} : \text{radio} \leq 120$$

4.4.2 Code



The screenshot displays the AMPL IDE interface. On the left, the 'EXPLORER' pane shows a project named 'AMPL MODELS' containing several files: 'group10_HW1_p3.txt', 'group10_HW1_p4a.txt', 'group10_HW1_p4b.txt', 'group10_HW1_p4c.txt', and 'group10_HW1_p4d.txt'. The main editor window is open to 'group10_HW1_p4d.txt', showing a linear programming model. The model is structured as follows:

```
1 Problem 4(c) - Hw1
2 # Advertising Campaign - Medium Selection
3 # Daniel Carpenter and Effauehi Moody
4
5 # MODEL SETUP -----
6
7 ## Reset Environment
8 reset;
9
10 ## Set up options and the solver
11 option solver cplex;
12 option cplex_options 'sensitivity';
13
14
15 # DECISION VARIABLES -----
16
17 ## tv = the number of minutes to air advertising on the television medium
18 var tv >= 0;
19
20 ## magazine = the number of pages to to advertise on the magazine medium
21 var magazine integer >= 0;
22
23 ## radio = the number of minutes to air advertising on the radio medium
24 var radio >= 0;
25
26
27 # OBJECTIVE -----
28
29 ## Objective is to Maximize the total audience reach
30 maximize audienceReach: (1.8*tv) + (1.0*magazine) + (0.25*radio);
31
32
33 # CONSTRAINTS -----
34
35 ## C1: Must not Exceed Budget of 1 Million dollars
36 subject to budget: (20000*tv) + (10000*magazine) + (2000*radio) <= 1000000;
37
38 ## C2: Must have at Least 10 minutes of air time on the TV medium
39 subject to minTimeTV: tv >= 10;
40
41 ## C3: Only 100 person weeks available, given it takes three weeks and one
42 ## week to create a tv and magazine minute for advertisement, respectively.
43 subject to personWeeks: (3*tv) + (1*magazine) + (1/7*radio) <= 100;
44
45 ## C4: Must sign up for at Least 2 magazine pages
46 subject to minMagazines: magazine >= 2;
47
48 ## C5: Must to exceed 120 minutes of radio
49 subject to maxRadio: radio <= 120;
50
51 # SOLVE -----
52
53 ## Solve the model
54 solve;
55
56 printf '\nPROBLEM 4(d) SOLUTIONS ----- \n\n';
57
58 ## Print the Objective
59 print 'Optimal / Maximum Audience Reach in Millions: ';
60 display audienceReach;
61
62 ## Print the decision variables
63 print 'Number of minutes dedicated to TV, and pages of Magazines: ';
64 display tv, magazine, radio;
65
66 ## Print the Budget Used
67 print 'Total Budget Used (US Dollars): ';
68 display 20000*tv + 10000*magazine + 2000*radio;
```

The bottom of the IDE shows a sidebar with 'OUTLINE' and 'TIMELINE' tabs, both of which are currently empty.

4.4.3 Output

```
ampl: model group10_HW1_p4d.txt
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal integer solution; objective 100.75
1 MIP simplex iterations
0 branch-and-bound nodes

suffix up OUT;
suffix down OUT;
suffix current OUT;

PROBLEM 4(d) SOLUTIONS -----

Optimal / Maximum Audience Reach in Millions:
audienceReach = 100.75

Number of minutes dedicated to TV, and pages of Magazines:
tv = 10
magazine = 53
radio = 119

Total Budget Used (US Dollars):
20000*tv + 10000*magazine + 2000*radio = 968000
```

5 Problem 5

5.1 Base Mathematical Formulation and Code

- *Each task shows a separate change to the base model. Therefore, each change should not accumulate.*

5.1.1 Mathematical Formulation

Given: P , a set of products
 a_j = tons per hour of product j , for each $j \in P$
 b = hours available at the mill
 c_j = profit per ton of product j , for each $j \in P$
 u_j = maximum tons of product j , for each $j \in P$

Define variables: X_j = tons of product j to be made, for each $j \in P$

Maximize: $\sum_{j \in P} c_j X_j$

Subject to: $\sum_{j \in P} (1/a_j) X_j \leq b$
 $0 \leq X_j \leq u_j$, for each $j \in P$

5.1.2 Code for Model .mod and Input Data .dat

The screenshot shows the Visual Studio Code interface with the AMPL Model (.mod) and Input Data (.dat) files open. The Explorer on the left lists the files in the 'group10_HW1_p5base' directory. The main editor displays the code for 'group10_HW1_p5base.mod' and 'group10_HW1_p5base.dat'.

group10_HW1_p5base.mod

```

1 # Problem 5
2 # BASE MODEL
3 # Steel MILL
4 # Daniel Carpenter and Effauehi Moody
5
6 # MODEL SETUP -----
7
8 ## Reset Environment
9 reset;
10
11 ## Set up options and the solver
12 option solver cplex;
13 option cplex_options 'sensitivity';
14
15
16 # SETS -----
17 set PROD; # products
18 set STAGE; # stages
19
20 # PARAMETERS -----
21 param rate {PROD,STAGE} > 0; # tons per hour in each stage
22 param avail {STAGE} >= 0; # hours available/week in each stage
23 param profit {PROD}; # profit per ton
24
25 param commit {PROD} >= 0; # lower limit on tons sold in week
26 param market {PROD} >= 0; # upper limit on tons sold in week
27
28
29 # DECISION VARS -----
30 var Make {p in PROD} >= commit[p], <= market[p]; # tons produced
31
32
33 # OBJECTIVE: total profits from all products -----
34 maximize Total_Profit: sum {p in PROD} profit[p] * Make[p];
35
36
37 # CONSTRAINTS -----
38 # In each stage: total of hours used by all
39 # products may not exceed hours available
40 subject to Time {s in STAGE}: sum {p in PROD} (1/rate[p,s]) * Make
41 [p] <= avail[s];
42
43 # DATA INPUTS -----
44 data group10_HW1_p5base.dat;
45
46 # SOLVE -----
47 solve;
48 display Make;

```

group10_HW1_p5base.dat

```

1 # Problem 5
2 # BASE DATA FILE
3 # Steel MILL
4 # Daniel Carpenter and Effauehi Moody
5
6 # SETS -----
7 set PROD := bands coils plate;
8 set STAGE := reheat roll;
9
10 # PARAMETERS -----
11 param rate: reheat roll :=
12     bands 200 200
13     coils 200 140
14     plate 200 160 ;
15
16 param: profit commit market :=
17     bands 25 1000 6000
18     coils 30 500 4000
19     plate 29 750 3500 ;
20
21 param avail :=
22     reheat 35
23     roll 40 ;

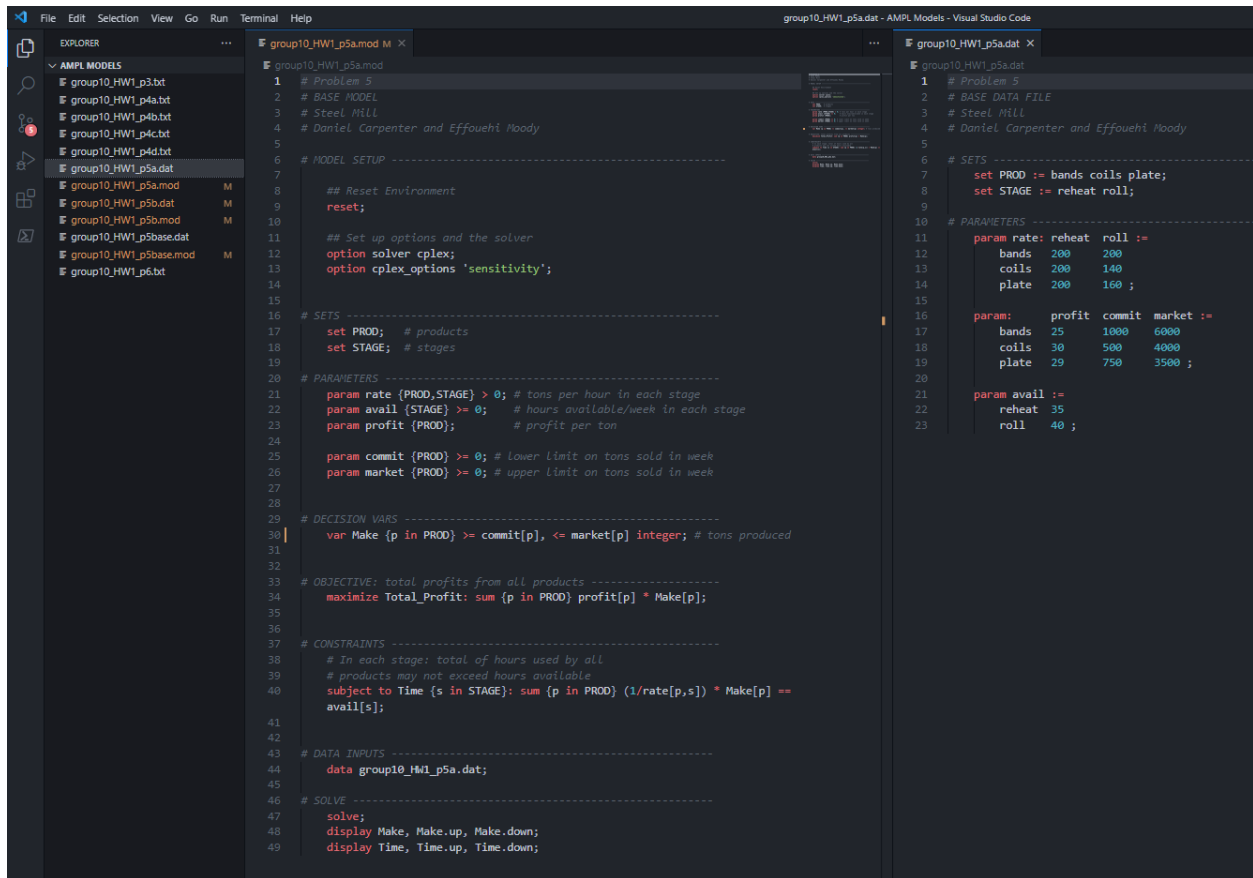
```

5.2 Task a

5.2.1 Changed Constraint for Total Hours

- *Change the constraints so that total hours used by all products must equal the total hours available for each stage*

5.2.2 Code



```
File Edit Selection View Go Run Terminal Help
group10_HWI_p5a.dat - AMPL Models - Visual Studio Code

EXPLORER
  AMPL MODELS
    group10_HWI_p3a.mod
    group10_HWI_p4a.mod
    group10_HWI_p4b.mod
    group10_HWI_p4c.mod
    group10_HWI_p4d.mod
    group10_HWI_p5a.mod
    group10_HWI_p5b.mod
    group10_HWI_p5c.mod
    group10_HWI_p5d.mod
    group10_HWI_p5e.mod
    group10_HWI_p6a.mod

group10_HWI_p5a.mod
1 # Problem 5
2 # BASE MODEL
3 # Steel Mill
4 # Daniel Carpenter and Effouehi Moody
5
6 # MODEL SETUP -----
7
8 ## Reset Environment
9 reset;
10
11 ## Set up options and the solver
12 option solver cplex;
13 option cplex_options 'sensitivity';
14
15
16 # SETS -----
17 set PROD; # products
18 set STAGE; # stages
19
20 # PARAMETERS -----
21 param rate {PROD,STAGE} >= 0; # tons per hour in each stage
22 param avail {STAGE} >= 0; # hours available/week in each stage
23 param profit {PROD}; # profit per ton
24
25 param commit {PROD} >= 0; # lower limit on tons sold in week
26 param market {PROD} >= 0; # upper limit on tons sold in week
27
28
29 # DECISION VARS -----
30 var Make {p in PROD} >= commit[p], <= market[p] integer; # tons produced
31
32
33 # OBJECTIVE: total profits from all products -----
34 maximize Total_Profit: sum {p in PROD} profit[p] * Make[p];
35
36
37 # CONSTRAINTS -----
38 # In each stage: total of hours used by all
39 # products may not exceed hours available
40 subject to Time {s in STAGE}: sum {p in PROD} (1/rate[p,s]) * Make[p] ==
    avail[s];
41
42
43 # DATA INPUTS -----
44 data group10_HWI_p5a.dat;
45
46 # SOLVE -----
47 solve;
48 display Make, Make.up, Make.down;
49 display Time, Time.up, Time.down;
```

```
group10_HWI_p5a.dat
1 # Problem 5
2 # BASE DATA FILE
3 # Steel Mill
4 # Daniel Carpenter and Effouehi Moody
5
6 # SETS -----
7
8 set PROD := bands coils plate;
9
10 # PARAMETERS -----
11
12 param rate: rehear roll :=
13   bands 200 200
14   coils 200 140
15   plate 200 160 ;
16
17 param: profit commit market :=
18   bands 25 1000 6000
19   coils 30 500 4000
20   plate 29 750 3500 ;
21
22 param avail :=
23   rehear 35
24   roll 40 ;
```


5.2.3 Output

There is no difference in the optimal solution because the range of Time before there is a change in optimal remains the same, and the hours available have not changed.

```
ampl: model group10_HW1_p5base.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 190071.4286
2 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
ampl: model group10_HW1_p5a.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 190071.4286
2 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
:      Make      Make.up      Make.down      :=
bands  3357.14      27.6      -1e+20
coils   500        31.8571    -1e+20
plate  3142.86     1e+20      27.9167
;

:      Time      Time.up      Time.down      :=
reheat 1800      37.6429     34.5536
roll   3200      40.4464     37.0089
;
```

5.3 Task b

5.3.1 New Constraint for Max Weight

- Restrict the total weight of all products to be less than a new parameter, $max_weight = 6,500$

$$totalWeight : \sum_{p \in PROD} Make_p \leq max_weight$$

5.3.2 Code

```

1 # Problem 5
2 # BASE MODEL
3 # Steel MILL
4 # Daniel Carpenter and Effauehi Moody
5
6 # MODEL SETUP -----
7
8 ## Reset Environment
9 print;
10 reset;
11
12 ## Set up options and the solver
13 option solver cplex;
14 option cplex_options 'sensitivity';
15
16
17 # SETS -----
18 set PROD; # products
19 set STAGE; # stages
20
21 # PARAMETERS -----
22 param rate {PROD,STAGE} > 0; # tons per hour in each stage
23 param profit {PROD}; # profit per ton
24 param commit {PROD} >= 0; # lower limit on tons sold in week
25 param market {PROD} >= 0; # upper limit on tons sold in week
26 param avail {STAGE} >= 0; # hours available/week in each stage
27 param max_weight >= 0; # Max weight/week
28
29 # DECISION VARS -----
30 var Make {p in PROD} >= commit[p], <= market[p] integer; # tons produced
31
32
33 # OBJECTIVE: total profits from all products -----
34 maximize Total_Profit: sum {p in PROD} profit[p] * Make[p];
35
36
37 # CONSTRAINTS -----
38 # In each stage: total of hours used by all
39 # products may not exceed hours available
40 subject to Time
41 {s in STAGE}:
42 sum {p in PROD}
43 (1/rate[p,s]) * Make[p] <= avail[s];
44
45 # Constraint 2: Total weight of all products Less than 6500 tons
46 subject to Total_Weight:
47 sum {p in PROD} Make[p] <= max_weight;
48
49
50 # DATA INPUTS -----
51 data group10_HW1_p5b.dat;
52
53 # SOLVE -----
54 solve;
55 display Make;

```

5.3.3 Output

The total number of tons has reduced from 7,000 to 6,500 per week

```
ampl: model group10_HW1_p5base.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 190071.4286
2 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
Make [*] :=
bands 3357.14
coils 500
plate 3142.86
;

ampl: model group10_HW1_p5b.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 183791.6667
3 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
Make [*] :=
bands 1541.67
coils 1458.33
plate 3500
;
```

5.4 Task c

5.4.1 Changed Objective Function

- *Change the objective function to maximize total tons*

$$\text{maximize } Total_Tons = \sum_{p \in PROD} Make_p$$

5.4.2 Code

```
1 # Problem 5
2 # BASE MODEL
3 # Steel Mill
4 # Daniel Carpenter and Effouehi Moody
5
6 # MODEL SETUP -----
7
8 ## Reset Environment
9 reset;
10
11 ## Set up options and the solver
12 option solver cplex;
13 option cplex_options 'sensitivity';
14
15 # SETS -----
16 set PROD; # products
17 set STAGE; # stages
18
19 # PARAMETERS -----
20 param rate {PROD,STAGE} > 0; # tons per hour in each stage
21 param avail {STAGE} >= 0; # hours available/week in each stage
22 param profit {PROD}; # profit per ton
23
24 param commit {PROD} >= 0; # Lower limit on tons sold in week
25 param market {PROD} >= 0; # upper limit on tons sold in week
26
27 # DECISION VARS -----
28 var Make {p in PROD} >= commit[p], <= market[p]; # tons produced
29
30 # OBJECTIVE: total TONS from all products -----
31 maximize Total_Tons: sum {p in PROD} Make[p];
32
33 # CONSTRAINTS -----
34 # In each stage: total of hours used by all
35 # products may not exceed hours available
36 subject to Time {s in STAGE}: sum {p in PROD} (1/rate[p,s]) * Make[p] <= avail
37 [s];
38
39 # DATA INPUTS -----
40 data group10_HW1_p5c.dat;
41
42 # SOLVE -----
43 solve;
44 display Make;
```

5.4.3 Output

The data file does not make a difference in the optimal (assuming that is what the question is asking). Please note that the total number of tons produced are the same as in the **base** model; however, the allocation of tons have shifted among each of the products.

```
ampl: model group10_HW1_p5base.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 190071.4286
2 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
Make [*] :=
bands 3357.14
coils 500
plate 3142.86
;

ampl: model group10_HW1_p5c.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 7000
1 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
Make [*] :=
bands 5750
coils 500
plate 750
;
```

5.5 Task d

5.5.1 New Constraint

- *Minimum Share of Tons for each Product*

$$Share_of_Products : Make_j \geq share_j \times \sum_{k \in PROD} Make_k, \quad \forall j \in PROD$$

5.5.2 Code (Part I)

```
1 # Problem 5
2 # BASE MODEL
3 # Steel Mill
4 # Daniel Carpenter and Effouehi Hoody
5
6 # MODEL SETUP -----
7
8 ## Reset Environment
9 reset;
10
11 ## Set up options and the solver
12 option solver cplex;
13 option cplex_options 'sensitivity';
14
15 # SETS -----
16
17 set PROD; # products
18 set STAGE; # stages
19
20 # PARAMETERS -----
21
22 param rate {PROD,STAGE} >= 0; # tons per hour in each stage
23 param avail {STAGE} >= 0; # hours available/week in each stage
24 param profit {PROD}; # profit per ton
25
26 param share {PROD} >= 0; # Lower limit on tons sold in week
27 param market {PROD} >= 0; # upper limit on tons sold in week
28
29 # DECISION VARS -----
30
31 var Make {p in PROD} <= market[p]; # tons produced
32
33 # OBJECTIVE: total profits from all products -----
34 maximize Total_Profit: sum {p in PROD} profit[p] * Make[p];
35
36 # CONSTRAINTS -----
37
38 # In each stage: total of hours used by all
39 # products may not exceed hours available
40 subject to Time {s in STAGE}: sum {p in PROD} (1/rate[p,s]) * Make[p] <= avail[s];
41
42 # Minimum Share of Tons for each Product
43 subject to Share_of_Products {j in PROD}:
44     Make[j] >= share[j] * sum {k in PROD} Make[k];
45
46 # DATA INPUTS -----
47
48 data group10_HW1_p5d.dat;
49
50 # SOLVE -----
51 solve;
52 display Make;
```

```
1 # Problem 5
2 # BASE DATA FILE
3 # Steel Mill
4 # Daniel Carpenter and Effouehi Hoody
5
6 # SETS -----
7
8 set PROD := bands coils plate;
9 set STAGE := rehear roll;
10
11 # PARAMETERS -----
12
13 param rate: rehear roll :=
14     bands 200 200
15     coils 200 140
16     plate 200 160 ;
17
18 param: profit share market :=
19     bands 25 0.4 8000
20     coils 30 0.4 4000
21     plate 29 0.1 3500 ;
22
23 param avail :=
24     rehear 35
25     roll 40 ;
```

5.5.3 Output (Part I)

Note that bands represent ~49.99%, coils: 40%, and plates: 10%

```
ampl: model group10_HW1_p5d.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 183211.9403
5 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
Make [*] :=
bands 3343.28
coils 2674.63
plate 668.657
;
```

5.5.4 Code (Part II)

```
1 # Problem 5
2 # BASE MODEL
3 # Steel MILL
4 # Daniel Carpenter and Effaouhi Moody
5
6 # MODEL SETUP -----
7
8 ## Reset Environment
9 reset;
10
11 ## Set up options and the solver
12 option solver cplex;
13 option cplex_options 'sensitivity';
14
15 # SETS -----
16
17 set PROD; # products
18 set STAGE; # stages
19
20 # PARAMETERS -----
21
22 param rate (PROD,STAGE) >= 0; # tons per hour in each stage
23 param avail (STAGE) >= 0; # hours available/week in each stage
24 param profit (PROD); # profit per ton
25
26 param share (PROD) >= 0; # Lower Limit on tons sold in week
27 param market (PROD) >= 0; # upper limit on tons sold in week
28
29 # DECISION VARS -----
30
31 var Make {p in PROD} <= market[p]; # tons produced
32
33 # OBJECTIVE: total profits from all products -----
34
35 maximize Total_Profit: sum {p in PROD} profit[p] * Make[p];
36
37 # CONSTRAINTS -----
38
39 # In each stage: total of hours used by all
40 # products may not exceed hours available
41
42 subject to Time {s in STAGE}: sum {p in PROD} (1/rate[p,s]) * Make[p] <= avail[s];
43
44 # Minimum Share of Tons For each Product
45
46 subject to Share_of_Products {j in PROD}:
47     Make[j] >= share[j] * sum {k in PROD} Make[k];
48
49 # DATA INPUTS -----
50
51 data group10_HW1_p5d.dat;
52
53 # SOLVE -----
54
55 solve;
56 display Make;
```

5.5.5 Output (Part II)

Profit is zero because it is impossible for bands to reach 50% of the share.

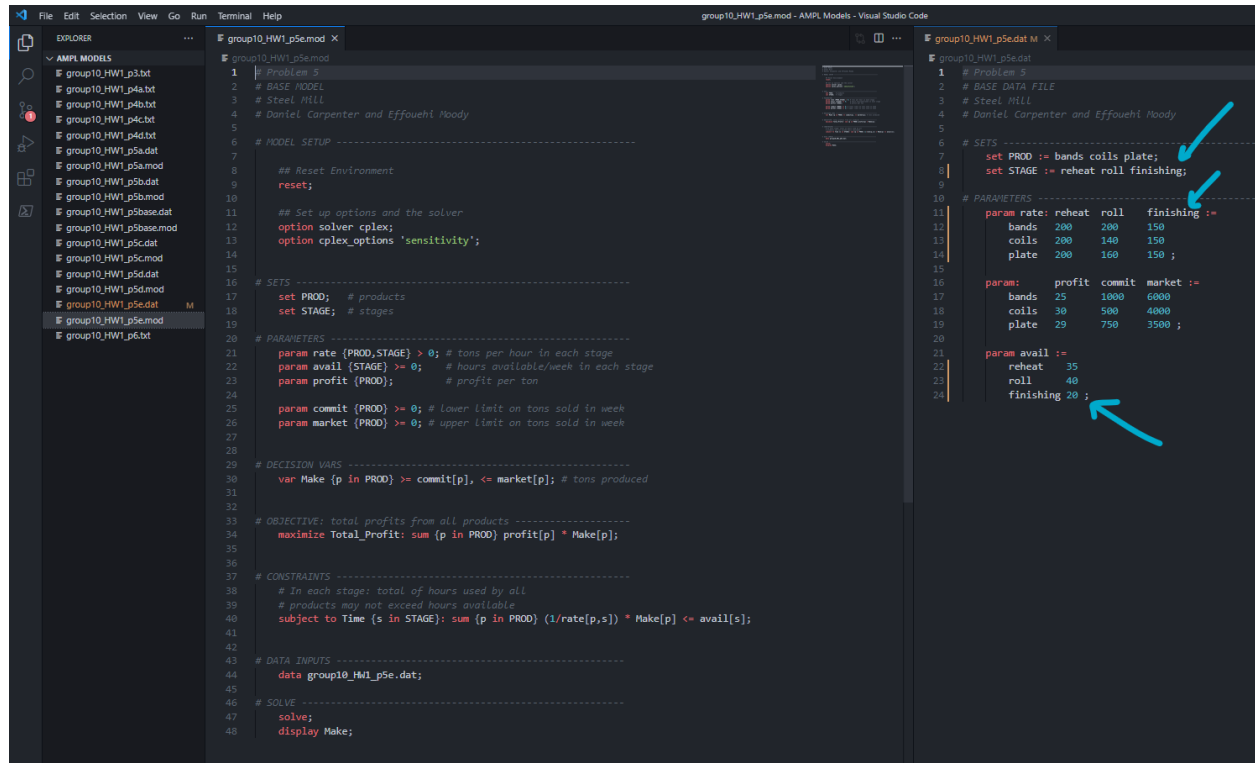
```
ampl: model group10_HW1_p5d.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 0
4 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
Make [*] :=
bands 0
coils 0
plate 0
;
```


5.6 Task e

5.6.1 Changing Input Data via .dat File

Simply add the new item within the set called `finishing`, then add the its the associate values to the `rate` and `avail` parameters.



```
1 # Problem 5
2 # BASE MODEL
3 # Steel MILL
4 # Daniel Carpenter and Effouehi Moody
5
6 # MODEL SETUP -----
7
8 ## Reset Environment
9 reset;
10
11 ## Set up options and the solver
12 option solver cplex;
13 option cplex_options 'sensitivity';
14
15 # SETS -----
16
17 set PROD; # products
18 set STAGE; # stages
19
20 # PARAMETERS -----
21
22 param rate {PROD, STAGE} >= 0; # tons per hour in each stage
23 param avail {STAGE} >= 0; # hours available/week in each stage
24 param profit {PROD}; # profit per ton
25
26 param commit {PROD} >= 0; # Lower limit on tons sold in week
27 param market {PROD} >= 0; # upper limit on tons sold in week
28
29 # DECISION VARS -----
30
31 var Make {p in PROD} >= commit[p], <= market[p]; # tons produced
32
33 # OBJECTIVE: total profits from all products -----
34 maximize Total_Profit: sum {p in PROD} profit[p] * Make[p];
35
36 # CONSTRAINTS -----
37
38 # In each stage: total of hours used by all
39 # products may not exceed hours available
40 subject to Time {s in STAGE}: sum {p in PROD} (1/rate[p,s]) * Make[p] <= avail[s];
41
42 # DATA INPUTS -----
43
44 data group10_HW1_p5e.dat;
45
46 # SOLVE -----
47
48 solve;
49 display Make;
```

```
1 # Problem 5
2 # BASE DATA FILE
3 # Steel MILL
4 # Daniel Carpenter and Effouehi Moody
5
6 # SETS -----
7
8 set PROD := bands coils plate;
9 set STAGE := reheat roll finishing;
10
11 # PARAMETERS -----
12
13 param rate: reheat roll finishing :=
14   bands 200 200 150
15   coils 200 140 150
16   plate 200 160 150 ;
17
18 param: profit commit market :=
19   bands 25 1000 6000
20   coils 30 500 4000
21   plate 29 750 3500 ;
22
23 param avail :=
24   reheat 35
25   roll 40
26   finishing 20 ;
```

5.6.2 Output

```
ampl: model group10_HW1_p5e.mod
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 84250
0 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
Make [*] :=
bands 1000
coils 1250
plate 750
;
```

6 Problem 6

6.1 Task a - c

6.1.1 Decision Variables

bondA: dollars $\in \mathbb{R}$ to invest in bond A

bondB: dollars $\in \mathbb{R}$ to invest in bond B

bondC: dollars $\in \mathbb{R}$ to invest in bond C

bondD: dollars $\in \mathbb{R}$ to invest in bond D

bondE: dollars $\in \mathbb{R}$ to invest in bond E

6.1.2 Objective Function

- Maximize the Expected Earnings of the portfolio

$$\text{Maximize } Z = (0.043 \times \text{bondA}) + (0.027 \times \text{bondB}) + (0.025 \times \text{bondC}) + (0.022 \times \text{bondD}) + (0.045 \times \text{bondE})$$

6.1.3 Constraints

C1: Budget to invest is \$10 MM or less

$$\text{budget} : \text{bondA} + \text{bondB} + \text{bondC} + \text{bondD} + \text{bondE} \leq 10$$

C2: At least \$4 million must be invested in government and agency bonds

$$\text{govtAndAgency} : \text{bondB} + \text{bondC} + \text{bondD} \geq 4$$

C3: Average Quality of the Portfolio must not exceed 1.4

$$\text{avgQuality} : (0.6 \times \text{bondA}) + (0.6 \times \text{bondB}) - (0.4 \times \text{bondC}) - (0.4 \times \text{bondD}) + (3.6 \times \text{bondE}) \leq 0$$

C4: The Average Maturity must not Exceed Five Years

$$\text{avgMaturity} : (4 \times \text{bondA}) + (10 \times \text{bondB}) - (1 \times \text{bondC}) - (2 \times \text{bondD}) - (3 \times \text{bondE}) \leq 0$$

C5: Only select Bonds A and D (Don't select B, C, or E)

$$\text{onlyAandB} : \text{bondB} + \text{bondC} + \text{bondE} = 0;$$

C6: Municipal Bonds must be less than or equal to \$3 MM

$$\text{municipal} : \text{bondA} \leq 3;$$

6.1.4 Code

```
File Edit Selection View Go Run Terminal Help
EXPLORER
  AMPL MODELS
    group10_HW1_p3.txt M
    group10_HW1_p4a.txt
    group10_HW1_p4b.txt
    group10_HW1_p4c.txt
    group10_HW1_p4d.txt
    group10_HW1_p6.txt U
  group10_HW1_p6.txt
1  Problem 6 - HW1
2  # Portfolio Constraint Problem Revisited
3  # Daniel Carpenter and Effaouhi Moody
4
5  # MODEL SETUP -----
6
7  ## Reset Environment
8  reset;
9
10 ## Set up options and the solver
11 option solver cplex;
12 option cplex_options 'sensitivity';
13
14
15 # DECISION VARIABLES -----
16
17 ## Dollars Invested in Respective Bonds (A, B, C, D, E)
18 var bondA >= 0;
19 var bondB >= 0;
20 var bondC >= 0;
21 var bondD >= 0;
22 var bondE >= 0;
23
24
25 # OBJECTIVE -----
26
27 ## Objective is to Maximize the Expected Earnings of the portfolio.
28 ## Coefficients resemble the expected return on the bonds, in the form of decimal.
29 maximize expectedEarnings: 0.043*bondA + 0.027*bondB + 0.025*bondC + 0.022*bondD + 0.045*bondE;
30
31
32 # CONSTRAINTS -----
33
34 ## Constraint 1: Budget to Invest is $10 MM or less
35 subject to budget: bondA + bondB + bondC + bondD + bondE <= 10;
36
37 ## Constraint 2: At Least $4 million must be invested in government and agency bonds
38 subject to govtAndAgency: bondB + bondC + bondD >= 4;
39
40 ## Constraint 3: Average Quality of the Portfolio must not exceed 1.4
41 subject to avgQuality: 0.6*bondA + 0.6*bondB - 0.4*bondC - 0.4*bondD + 3.6*bondE <= 0;
42
43 ## Constraint 4: The Average Maturity must not Exceed Five Years
44 subject to avgMaturity: 4*bondA + 10*bondB - bondC - 2*bondD - 3*bondE <= 0;
45
46 ## Constraint 5: Only select Bonds A and D (Don't select B, C, or E)
47 subject to onlyAandB: bondB + bondC + bondE == 0;
48
49 ## Constraint 6: Municipal Bonds must be less than or equal to $3 MM
50 subject to municipal: bondA <= 3;
51
52
53 # SOLVE -----
54
55 ## Solve the model
56 solve;
57
58 print;
59 print 'Objective: Optimal Solution for Expected Earnings (millions of dollars)';
60 display expectedEarnings;
61
62 print 'Decision Variables for Bond A, B, C, D, and E, Respectively';
63 display bondA, bondD;
64
65 print 'Shadow Price of the municipal limit: ';
66 display municipal;
67
68 print 'Range of the Municipal Limit before it becomes infeasible';
69 print '(assuming nothing else changes)';
70 display municipal.up, municipal.down;
71
72 print 'You may not borrow more than 2.83%, since that is the expected';
73 print 'yield to maturity (38% of bondA * 4.3%) + (78% of bondD * 2.2%)';
74
75 print;
76 print 'If you borrowed at a rate greater than the expected YTM, then the)';
77 print 'venture would not be profitable.';
78
79
```

6.1.5 Output

```
ampl: model group10_HW1_p6.txt
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 0.283
0 dual simplex iterations (0 in phase I)

suffix up OUT;
suffix down OUT;
suffix current OUT;
|
Objective: Optimal Solution for Expected Earnings (millions of dollars)
expectedEarnings = 0.283

Decision Variables for Bond A, B, C, D, and E, Respectively
bondA = 3
bondD = 7

Shadow Price of the municipal limit:
municipal = 0.021

Range of the Municipal Limit before it becomes infeasible
(assuming nothing else changes)
municipal.up = 0
municipal.down = 0
```

6.2 Task d:

You may not borrow more than 2.83%, since that is the expected yield to maturity $(30\% \text{ of bondA} * 4.3\%) + (70\% \text{ of bondD} * 2.2\%)$

6.3 Task e:

If you borrowed at a rate greater than the expected YTM, then the venture would not be profitable.