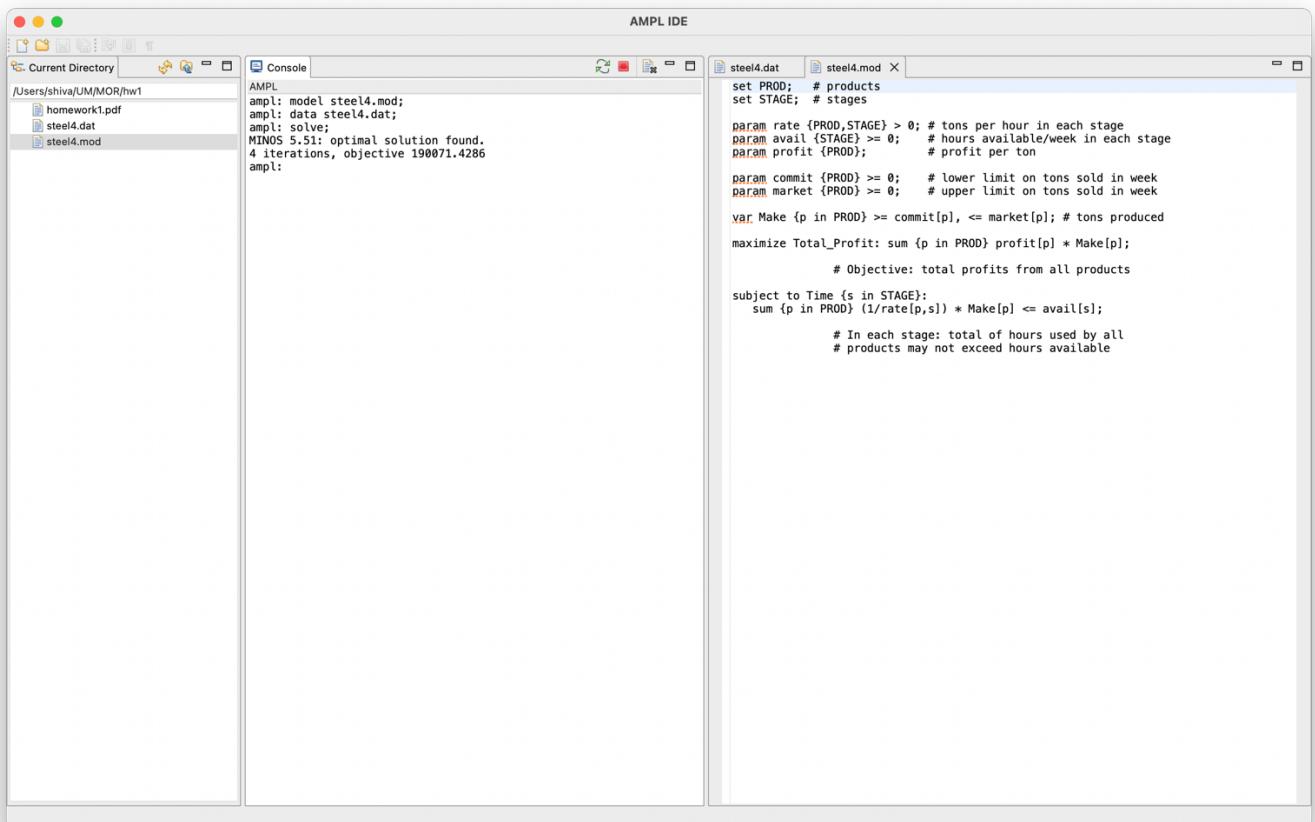
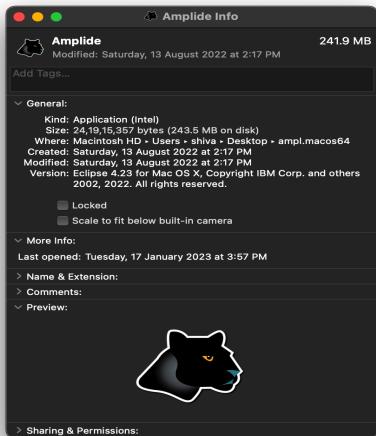


Homework-1

1.

a. AMPL Software installed on the local folder called Desktop



The optimal objective value should be reported as 190071.4286. accurately matches with the above AMPL prompt output.

Homework-1

2.

a.

```
AMPL
ampl: model steel4.mod;
ampl: data steel4.dat;
ampl: solve;
MINLP 5.31: optimal solution found.
0 iterations, objective 190071.4286
ampl: option solver cplex;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 190071.4286
0 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 3357.14 6000 0
coils 500 500 4000 -1.85714
plate 750 3142.86 3500 0
;

ampl: let profit["coils"]:=31;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 190571.4286
0 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 3357.14 6000 0
coils 500 500 4000 -0.857143
plate 750 3142.86 3500 0
;

ampl: let profit["coils"]:=32;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 191270.8333
1 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 4354.17 6000 0
coils 500 1895.83 4000 0
plate 750 750 3500 -0.0833333
;

ampl: let profit["coils"]:=33;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 193166.6667
0 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 4354.17 6000 0
coils 500 1895.83 4000 0
plate 750 750 3500 -0.666667
;

ampl: let profit["coils"]:=34;
```

```
AMPL
ampl: let profit["coils"]:=34;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 195062.5
0 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 4354.17 6000 3.55271e-15
coils 500 1895.83 4000 0
plate 750 750 3500 -1.25
;

ampl: let profit["coils"]:=35;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 196958.3333
0 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 4354.17 6000 3.55271e-15
coils 500 1895.83 4000 0
plate 750 750 3500 -1.83333
;

ampl: let profit["coils"]:=36;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 199455.3571
1 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 1348.21 6000 0
coils 500 4000 4000 0.285714
plate 750 750 3500 -2.25
;

ampl: let profit["coils"]:=37;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 203455.3571
0 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 1348.21 6000 0
coils 500 4000 4000 1.28571
plate 750 750 3500 -2.25
;

ampl: let profit["coils"]:=38;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; Objective 207455.3571
0 simplex iterations (0 in phase I)
ampl: display Make.lb Make.Make.ub Make.rc;
: Make.lb Make Make.ub Make.rc :=
bands 1000 1348.21 6000 0
coils 500 4000 4000 2.28571
plate 750 750 3500 -2.25
```

Homework-1

```

Current Directory: /Users/shiva/UM/MOR/hw1
ampl: display Make.lb, Make, Make.ub, Make.rc;
:
bands 1600 1348.21 4000 0
coils 500 4000 4000 0
plate 750 750 3500 -2.25
;

ampl: let profit["coils"]:=37;
ampl: solve;
CPLEX 20.1.0: optimal solution; objective 199455.3571
0 simplex iterations (0 in phase I)
ampl: display Make.lb, Make, Make.ub, Make.rc;
:
bands 1600 1348.21 4000 0
coils 500 4000 4000 0
plate 750 750 3500 -2.25
;

ampl: let profit["coils"]:=38;
ampl: solve;
CPLEX 20.1.0: optimal solution; objective 203455.3571
0 simplex iterations (0 in phase I)
ampl: display Make.lb, Make, Make.ub, Make.rc;
:
bands 1600 1348.21 4000 0
coils 500 4000 4000 0
plate 750 750 3500 -2.25
;

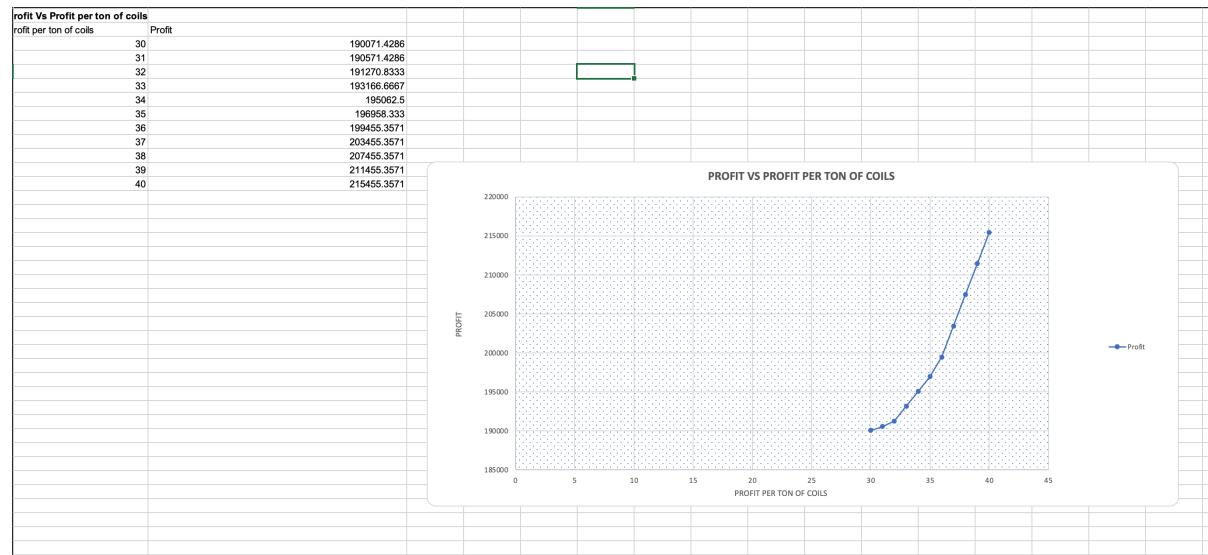
ampl: let profit["coils"]:=39;
ampl: solve;
CPLEX 20.1.0: optimal solution; objective 207455.3571
0 simplex iterations (0 in phase I)
ampl: display Make.lb, Make, Make.ub, Make.rc;
:
bands 1600 1348.21 4000 0
coils 500 4000 4000 0
plate 750 750 3500 -2.25
;

ampl: let profit["coils"]:=40;
ampl: solve;
CPLEX 20.1.0: optimal solution; objective 211455.3571
0 simplex iterations (0 in phase I)
ampl: display Make.lb, Make, Make.ub, Make.rc;
:
bands 1600 1348.21 4000 0
coils 500 4000 4000 0
plate 750 750 3500 -2.25
;

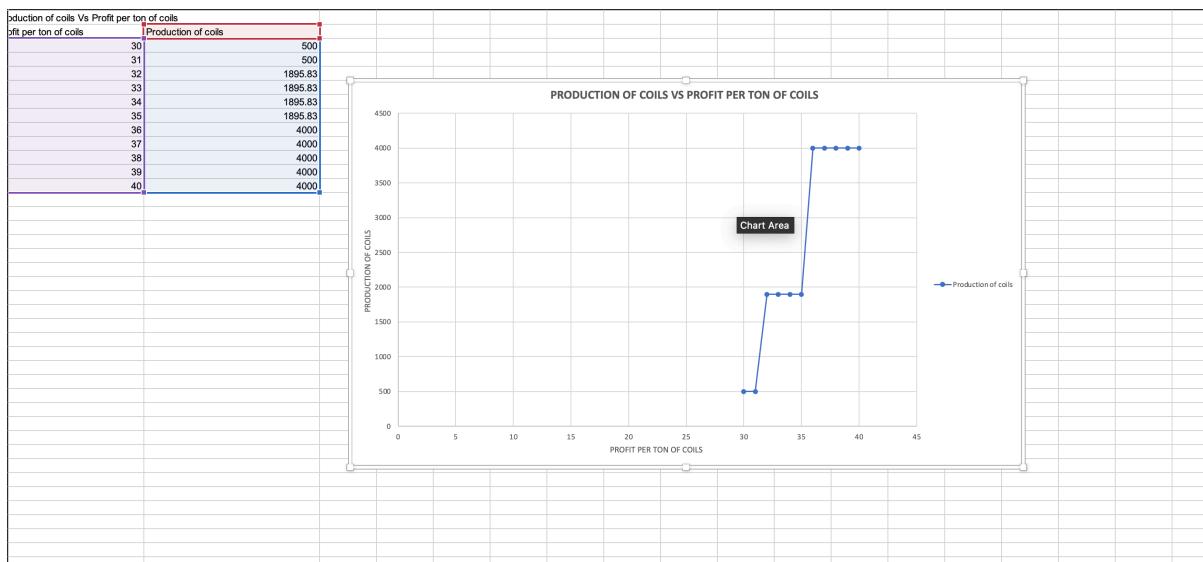
```

As we can see, the output of coils surpasses 500 and rises to 1895.83 following a \$2 increase in the profit per ton of coils. The manufacturing of coils now rises to 4000 after a \$4 increase. As a result, we see that the connection between coil production and profit per ton of coils is linear.

b) Solution:



Homework-1



3. Solution:

The screenshot shows the AMPL IDE interface. On the left, the 'Current Directory' pane lists files: adplan.dat, adplan.mod, homework1.pdf, hw1-Solutions.docx, steel4.dat, steel4.mod, and ~\$1-Solutions.docx. The central 'Console' pane shows the command-line session:

```
AMPL
ampl: model adplan.mod;
ampl: data adplan.dat;
ampl: solve;
CPLEX 20.1.0.0: optimal solution found.
3 iterations, objective 10960000
ampl: option solver cplex;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 10960000
0 simplex iterations (0 in phase I)
ampl: |
```

The right pane displays the contents of the 'adplan.mod' file:

```
param n; # Number of Decision Variables
param m; # Number of Constraints
set J := {1..n}; #set of decision variables
set I := {1..m}; #set of constraints

param C {} >= 0; #Coefficients of Objective Function
param A {I,J} >= 0; #Coefficients of Constraint Matrix
param B {I} >= 0; #LHS of the Constraints
param D {I} >= 0; #RHS of the constraints

var X {J} >= 0; #decision variables

maximize z: sum {j in J} C[j] * X[j]; #Objective Function
s.t. Constraint {i in I}:
    B[i] <= sum {j in J} A[i,j] * X[j] <= D[i]; # B <= Constraints <= D

# PLEASE RUN THE COMMAND include adplan.run; IN THE CONSOLE to obtain the r
```

Homework-1

The screenshot shows the AMPL IDE interface. On the left, the 'Current Directory' pane lists files: adspian.dat, adspian.mod, homework1.pdf, hw1-Solutions.docx, steel4.dat, steel4.mod, and -\$1-Solutions.docx. The 'Console' pane displays the command-line session:

```
AMPL
ampl: model adspian.mod;
ampl: data adspian.dat;
ampl: solve;
NLP: optimal solution found.
3 iterations, objective 10960000
ampl: option solver cplex;
ampl: solve;
CPLEX 20.1.0.0: optimal solution; objective 10960000
0 simplex iterations (0 in phase I)
ampl:
```

The right pane shows the contents of 'steel4.dat':

```
param n := 4;
param m := 7;

param C :=
  1 400000
  2 900000
  3 500000
  4 200000;

param A: 1 40000 75000 30000 15000
        2 300000 400000 200000 100000
        3 40000 75000 0 0
        4 1 0 0 0
        5 0 1 0 0
        6 0 0 1 0
        7 0 0 0 1;

param B :=
  1 0
  2 3000000
  3 0
  4 3
  5 2
  6 5
  7 5;

param D :=
  1 800000
  2 Infinity
  3 500000
  4 Infinity
  5 Infinity
  6 10
  7 10;
```

The optimal objective value should be reported as 10960000

Let the amount of advertising units purchased for radio, magazines, daytime television, and prime time television, respectively, be x_1 , x_2 , x_3 , and x_4 .

All potential clients who were contacted (in thousands)= $= 400x_1 + 900x_2 + 500x_3 + 200x_4$

These factors illustrate how the advertising budget is constrained=

$$40,000x_1 + 75,000x_2 + 30,000x_3 + 15,000x_4 \leq 800,000.$$

The limitation on the proportion of female clients who are reached by the advertising campaign is as follows= $300,000x_1 + 400,000x_2 + 200,000x_3 + 100,000x_4 \geq 2,00,000$.

the following are restrictions on television advertising= $40,000x_1 + 75,000x_2 \leq 500,000$

$$x_1 \geq 3$$

$$x_2 \geq 2.$$

Since radio and magazine advertising units should range from 5 to 10, the following restrictions apply

$$5 \leq x_3 \leq 10$$

$$5 \leq x_4 \leq 10.$$

The complete linear programming problem with some minor simplification

Is given below:

Homework-1

The full linear programming issue is provided below with some slight simplifications:

$$\text{Maximize} = Z = 400x_1 + 900x_2 + 500x_3 + 200x_4$$

$$\text{Subject to: } 40x_1 + 75x_2 + 30x_3 + 15x_4 \leq 800$$

$$30x_1 + 40x_2 + 20x_3 + 10x_4 \geq 200$$

$$40x_1 + 75x_2 \leq 500$$

$$x_1 \geq 3$$

$$x_2 \geq 2$$

$$x_3 \geq 5$$

$$x_3 \leq 10$$

$$x_4 \geq 5$$

$$x_4 \leq 10$$