Problem 1

a)

```
set DEPT;
set PROD;

var X{PROD} >=0;

param price{PROD};
param dem{PROD};
param cap{DEPT};
param a{PROD,DEPT};

maximize Revenue:
            sum{i in PROD} price[i]*X[i];

s.t. C1 {j in DEPT}:
            sum {i in PROD} a[i,j]*X[i] <= cap[j];

s.t. C2 {i in PROD}:
            X[i] <= dem[i];

data;

set DEPT := A B C D ;
set PROD := T1 T2 T3;
param price :=
  T1 10000  T2 12000 T3 15000;
param dem :=
  T1 45  T2 30 T3 40;
param cap :=
  A 1800  B 2300  C 2700  D 3000;
param a:
      A     B     C     D :=
  T1  12    30    20    35
  T2  16    10    40    26
  T3  25    17    18    22;
```

b) Modifications shown in yellow:

```
var Y{PROD} binary;

param fixedcost{PROD};

maximize Profit:  sum{i in PROD} price[i] * X[i]
                  - sum{i in PROD} fixedcost[i] * Y[i];

s.t. Constr2 {i in PROD}:
      X[i] <= dem[i] * Y[i];

data;
param fixedcost :=
      T1 300000  T2 400000  T3 500000;
```

c) Modifications shown in yellow:

```
var PRICE{PROD} >= 0;   # replaces parameter
var DEMAND{PROD} >= 0;  # replaces parameter

param alfa{PROD};
param beta{PROD};

maximize Revenue:
      sum{i in PROD} PRICE[i]*X[i];

s.t. Constr2 {i in PROD}:
      X[i] <= DEMAND[i];

s.t. Constr3 {i in PROD}:
      DEMAND[i] = alfa[i] - beta[i] * PRICE[i];

data;

param alfa :=
      T1 195      T2 270      T3 415;
param beta :=
      T1 0.015    T2 0.020    T3 0.025;
```

Problem 2

a)

```
set TYPE;

var X{TYPE} >=0;

param qi1{TYPE};
param qi2{TYPE};
param unitcost{TYPE};
param qi1_min;
param qi1_max;
param qi2_min;
param qi2_max;
param demand;

minimize TotalCosts:    sum{i in TYPE} unitcost[i] * X[i];

s.t. C1:    sum {i in TYPE} qi1[i] * X[i] >= qi1_min * demand;
s.t. C2:    sum {i in TYPE} qi1[i] * X[i] <= qi1_max * demand;
s.t. C3:    sum {i in TYPE} qi2[i] * X[i] >= qi2_min * demand;
s.t. C4:    sum {i in TYPE} qi2[i] * X[i] <= qi2_max * demand;
s.t. C5:    sum {i in TYPE} X[i] = demand;

data;
set TYPE          := T1 T2 T3 T4 T5 ;
param qi1         := T1 99   T2 70   T3 78   T4 91   T5 85  ;
param qi2         := T1 210  T2 335  T3 280  T4 265  T5 250 ;
param unitcost    := T1 48   T2 43   T3 58   T4 46   T5 54  ;
param qi1_min :=  85;
param qi1_max :=  90;
param qi2_min := 260;
param qi2_max := 285;
param demand  := 500;
```

b)

```
var Y{TYPE} binary;

s.t. C6 {i in TYPE}:
        X[i] <= demand * Y[i]; # forces binary variable to be 1
   when the chemical is used

s.t. C7: X['T4'] >= 60 * Y['T4'];
```

c) (Need C6 as in b))

```
s.t. C8: Y['T3'] <= 1 - Y['T1'];
```

d) (Need C6 as in b))

```
s.t. C9: Y['T4'] <= 2 - Y['T2'] - Y['T3'];
```

e) (Need C6 as in b))

```
s.t. C9: Y['T3'] >= Y['T4'] + Y['T5'] - 1;
```

Problem 3

Math model:

```
Max   V1*X1 + V2*X2 + V3*X3
s.t.
      W1*X1 + W2*X2 + W3*X3 <= 12
      Xi = 0/1 for all i
```

```
V1 = 13     W1 = 6
V2 = 14     W2 = 7
V3 = 17     W3 = 4
```

Solution to initial relaxed problem, where Xi <= 1 for all i:

```
Using       V1/W1 = 2,17      Ranked no.1
            V2/W2 = 2,00      Ranked no.2
            V3/W3 = 1,75      Ranked no.3
```

```
X1 = 1
X2 = 6/7
X3 = 0
```

```
Total value = 13 + 14*6/7 = 25 (upper bound)
Total weight= 6 + 6*6/7 = 12
```

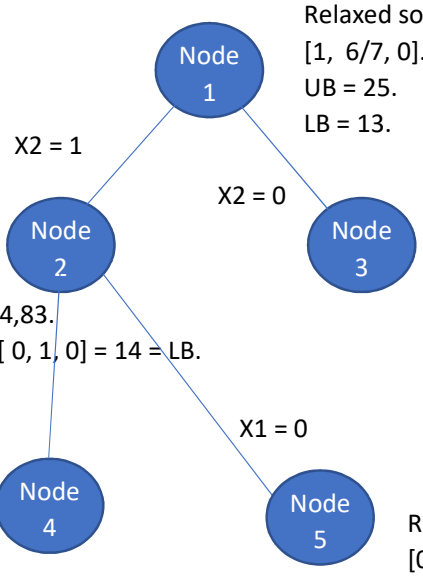
Constructing feasible solution by round the fractional value to 0, i.e.:

```
X1 = 1
X2 = 0
X3 = 0
```

```
Total value = 13 (lower bund)
Total weight= 6
```

Relaxed solution:
X1 = 1 X2 = 6/7 X3 = 0
Value of relaxed solution = 25 = UB
Feasible solution = [1, 0, 0]
LB = 13

Next, branch on X2



Relaxed solution:
[1, 6/7, 0].
UB = 25.
LB = 13.

Relaxed solution:
[5/6, 1, 0].
Value of relaxed solution = 13*5/6 + 14 = 24,83.
Value of feasible solution [0, 1, 0] = 14 = LB.

Relaxed solution:
[1, 0, 1] Value of relaxed solution = 13 + 7 = 20.
Relaxed solution is feasible, hence new LB = 20.
Search down this branch can be terminated.

Relaxed solution:
[1, 1, 0]
Total weight = 6+7= 13
⇨ Infeasible
Search down this branch can be terminated.

Relaxed solution:
[0, 1, 1] Value of relaxed solution =21.
Relaxed solution is feasible,
Search can be terminated.
Optimal solution is [0,1,1], value = 21