Cloud computing

a.

$$\begin{cases} \min\limits_{X} \sum_{i=1}^{1000} \sum_{j=1}^{20} c_{ij} X_{ij} \\ X_{ij} = 0 \text{ or } 1 \quad \text{for } 1 \le i \le 1000,\ 1 \le j \le 20 \\ \sum_{j=1}^{20} X_{ij} = 1 \quad \text{for } 1 \le i \le 1000 \\ \sum_{i=1}^{1000} X_{ij} \le C_j \quad \text{for } 1 \le j \le 20. \end{cases}$$

where c_{ij} is the energy cost, C_j is the capacity

X_{ij} is the variable

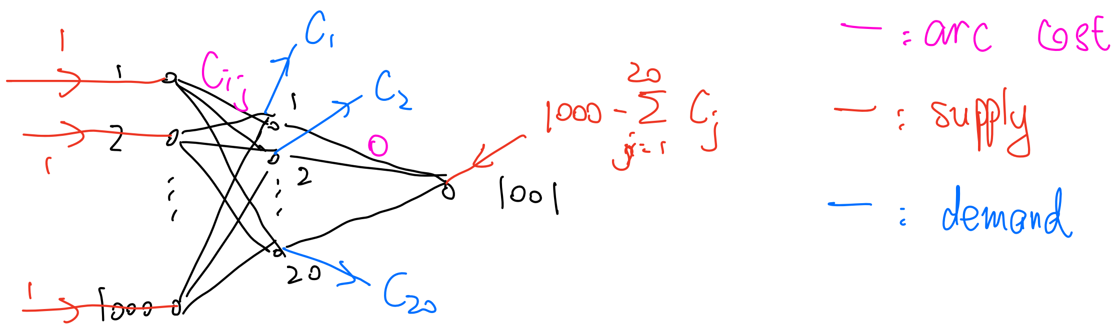
Yes it is a network flow,

we extend out 1000×20 variable matrix X

to 1001×20 variable matrix $\tilde{X}$

and extend energy cost $\tilde{c}_{ij} = \begin{cases} c_{ij} & \text{if } 1 \le i \le 1000 \\ 0 & \text{if } i = 1001 \end{cases}$

$$\begin{cases} \min\limits_{\tilde{X}} \sum_{i=1}^{1001} \sum_{j=1}^{20} \tilde{c}_{ij} \tilde{X}_{ij} \\ \sum_{j=1}^{20} \tilde{X}_{ij} = 1 \quad \text{for } 1 \le i \le 1000 \\ \left(\sum_{i=1}^{1000} \tilde{X}_{ij} \right) + \tilde{X}_{1001,j} = C_j \quad \text{for } 1 \le j \le 20 \\ \tilde{X}_{ij} \ge 0 \quad \text{for } 1 \le i \le 1001,\ 1 \le j \le 20. \end{cases}$$



by integral solution theorem, because 1, C_j are integral the solution of $\tilde{X}$ is integral, and thus X is integral.

b)

i) optimal energy: 41416

core codes:

```
model = Model(Gurobi.Optimizer)
@variable(model, X[1:1000, 1:20]);
@objective(model, Min, sum(sum(Menergy[i,j]*X[i,j] for j in 1:20) for i in 1:1000));
@constraint(model,a_constraint[i in 1:1000 ,j in 1:20],X[i,j] >= 0 );
@constraint(model,b_constraint[i in 1:1000],sum(X[i,j] for j in 1:20) == 1 );
@constraint(model,c_constraint[j in 1:20],sum(X[i,j] for i in 1:1000) <= Mcapacity[j] );
```

ii) optimal energy: 40883

Core codes:

```
model2 = Model(Gurobi.Optimizer)
@variable(model2, X2[1:1000, 1:20]);
@objective(model2, Min, sum(sum(Menergy[i,j]*X2[i,j] for j in 1:20) for i in 1:1000));
@constraint(model2,a_constraint[i in 1:1000 ,j in 1:20],X2[i,j] >= 0 );
@constraint(model2,b_constraint[i in 1:1000],sum(X2[i,j] for j in 1:20) == 1 );
#@constraint(model2,c_constraint[j in 1:20],sum(X2[i,j] for i in 1:1000) <= Mcapacity[j] );
```

✓ 1.3s

iii) We can compare the difference between solution in i) and ii)

number of jobs not assigned to lowest: 39

Core codes:

```
cnt = 0

for i in 1:100
    for j in 1:20
        cnt += abs(value(X[i,j]) - value(X2[i,j]))
    end
end
println(cnt/2)
```

✓ 0.6s

C.

$$\begin{cases} \min\limits_{X} \sum_{i=1}^{1000} \sum_{j=1}^{20} c_{ij} X_{ij} \\ X_{ij} = 0 \text{ or } 1 \quad \text{for} \quad 1 \le i \le 1000,\ 1 \le j \le 20 \\ \sum_{j=1}^{20} X_{ij} = 1 \quad \text{for} \quad 1 \le i \le 1000 \\ \sum_{i=1}^{1000} r_{ij} X_{ij} \le U_j \quad \text{for} \quad 1 \le j \le 20. \end{cases}$$

where C_{ij} is the energy cost, U_j is the maximal utilization
r_{ij} is the utilization of machine i performing job j
X_{ij} is the variable

No, it is not a network flow.

because equation $\sum_{i=1}^{1000} r_{ij} X_{ij} \le U_j$ for $1 \le j \le 20$,
the coefficients r_{ij} are not integral, it

is different from node equation in network flow.

Not guaranteed to be integer. because constraints $\sum_{i=1}^{1000} r_{ij} x_{ij} \le U_i$ will lead to non integral BFS.

d i) optimal energy: 46693

Core code:

```
model = Model(Gurobi.Optimizer)
@variable(model, X[1:1000, 1:20]>=0);
@objective(model, Min, sum(sum(Menergy[i,j]*X[i,j] for j in 1:20) for i in 1:1000));
@constraint(model,a_constraint[i in 1:1000 ,j in 1:20],X[i,j] >= 0 );
@constraint(model,b_constraint[i in 1:1000],sum(X[i,j] for j in 1:20) == 1 );
@constraint(model,c_constraint[j in 1:20],sum(Mutilization[i,j]*X[i,j] for i in 1:1000) <= Mmaxutil[j] );
```
✓ 4.0s

ii) optimal energy: 40883

Core code:

```
model2 = Model(Gurobi.Optimizer)
@variable(model2, X2[1:1000, 1:20]>=0);
@objective(model2, Min, sum(sum(Menergy[i,j]*X2[i,j] for j in 1:20) for i in 1:1000));
@constraint(model2,a_constraint[i in 1:1000 ,j in 1:20],X2[i,j] >= 0 );
@constraint(model2,b_constraint[i in 1:1000],sum(X2[i,j] for j in 1:20) == 1 );
#@constraint(model2,c_constraint[j in 1:20],sum(Mutilization[i,j]*X2[i,j] for i in 1:1000) <= Mmaxutil[j] );
```
✓ 0.9s

iii) we can compare the difference between solution in i) and ii) number of jobs not assigned to lowest:

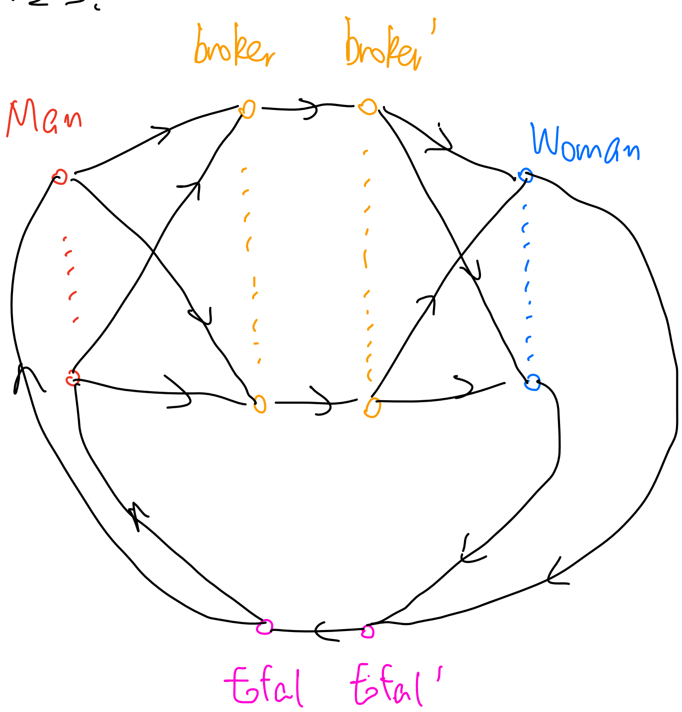
886

Core codes:

```
cnt = 0
for i in 1:1000
    temp = 0
    for j in 1:20
        temp += abs(value(X[i,j]) - value(X2[i,j]))
    end
    if temp > 0.001
        cnt += 1
    end
end

println(cnt)
```

Exercise 7.23.



We have $2(m+n+1)$ Nodes in total

They are Man Nodes: Man_i $i=1,\cdots,n$

Woman Nodes: $Woman_i$ $i=1,\cdots,n$

broker Nodes: $broker_i$ $i=1,\cdots,m$

broker' Nodes: $broker'_i$ $i=1,\cdots,m$

total Node, total' Node

Connectivity: $broker_i$ connect to $broker'_i$ $i=1,\cdots,m$

$broker_i$ connect to $\{Man_j \mid broker\ i \text{ knows man } j\}$ $i=1\cdots m$

$broker'_i$ connect to $\{Woman_j \mid broker\ i \text{ knows woman } j\}$ $i=1\cdots m$

Man_i connect to total $i=1,\cdots,n$

$Woman_i$ connect to total' $i=1,\cdots,n$

total connect to total'

Arc cost: Arc total $\to$ total' : cost $=-1$

other Arcs : cost $=0$

Arc capacity: $broker_i \to broker_i'$: capacity $\leq b_i$

$i=1,\cdots,m$

total $\to Man_i$: capacity ≤ 1

$i=1,\cdots,n$

$Woman_i \to$ total' : capacity ≤ 1

$i=1,\cdots,n$

Supply Node: None

demand Node: None

Exercise: 3.20.

(a). Current x must be feasible, so $\beta \geq 0$.

(b) $\beta < 0$

(c) $\beta \geq 0$, and ($\delta < 0$ or $\gamma < 0$ or $\xi < 0$)

(d) $\beta \geq 0$, and $\delta < 0$ and $\alpha \leq 0$, to make sure next step

we choose x_4,
we set
$\gamma > 0$, $\xi > 0$

(e) $\beta \geq 0$, and $\gamma < 0$, and $\eta > \frac{3}{4}$

(f) $\xi < 0$ and $\beta = 0$