

```
$Title ProjectScheduling_EC
```

**Set**

```
i /i1*i10/
```

**Alias** (i,j)

```
;
```

```
$ontext
```

```
Sets
```

```
P(i,j)
```

```
/
```

```
i1.i4
```

```
i2.i7
```

```
i3.i8
```

```
/
```

```
$offtext
```

**Parameters**

```
P(j,i)
```

```
;
```

**Set** M /m1\*m3/

**Parameters**

```
NM(i)
```

```
;
```

```
NM(i)=3;
```

```
;
```

**Set** t /t1\*t12/

```
;
```

**Parameters**

```
d(i,m)
```

```
c(i,m)
```

```
q(i,m)
```

```
fs(i,m,j)
```

```
TM
```

```
Qmin
```

```
B
```

```
w(i)
```

```
;
```

```
TM=card(T)
```

```
scalar lamda /0.7/;
```

```
Qmin=lamda*10*card(i);
```

```
$call GDXXRW ProjectSchedulingData.xlsx par=p rng=p! rdim=1 cdim=1
```

```
$gdxin ProjectSchedulingData.gdx
```

```
$load p
```

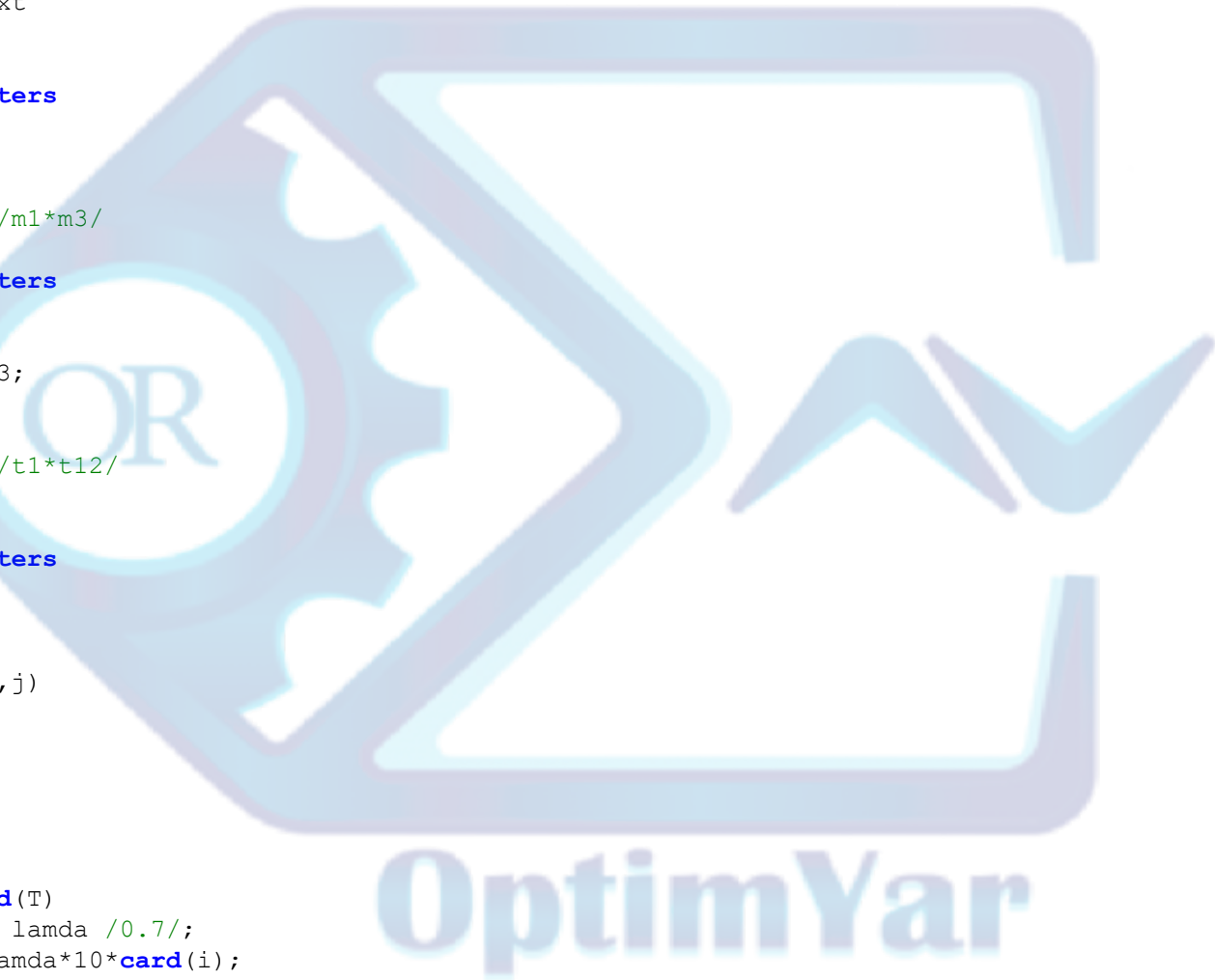
```
$gdxin
```

```
$call GDXXRW ProjectSchedulingData.xlsx par=d rng=d! rdim=1 cdim=1
```

```
$gdxin ProjectSchedulingData.gdx
```

```
$load d
```

```
$gdxin
```



```
$call GDXXRW ProjectSchedulingData.xlsx par=c rng=c! rdim=1 cdim=1
$gdxin ProjectSchedulingData.gdx
$load c
$gdxin
```

```
$call GDXXRW ProjectSchedulingData.xlsx par=q rng=q! rdim=1 cdim=1
$gdxin ProjectSchedulingData.gdx
$load q
$gdxin
```

```
$call GDXXRW ProjectSchedulingData.xlsx par=fs rng=fs! rdim=2 cdim=1
$gdxin ProjectSchedulingData.gdx
$load fs
$gdxin
```

```
$call GDXXRW ProjectSchedulingData.xlsx par=w rng=w! rdim=1 cdim=0
$gdxin ProjectSchedulingData.gdx
$load w
$gdxin
```

```
B=2*sum(i, smin(m, c(i, m)));
```

#### Display

```
p
d
c
q
B
fs
w
;
```

#### Binary Variable

```
x(i, m, t)
;
```

#### Free Variable

```
Cost
Time
Quality
;
Time.up=TM;
```

#### Equations

```
objT
objC
objQ
cons1
cons2
cons3
cons4
;
```

```
objT(i)..          time =g= sum({m, t}, (ord(t)+d(i, m)-1)*x(i, m, t));
```

OR

OptimYar

```

objC..          cost =e= sum({i,t,m},c(i,m)*x(i,m,t));
objQ..          quality =e= sum({i,t,m},w(i)*q(i,m)*x(i,m,t));
cons1(i)..      sum({m,t},x(i,m,t))=e=1;
cons2(i,j)$(p(j,i))..      sum({m,t},ord(t)*x(i,m,t)) =g= sum({m,t},(ord(t) +
m,t));          d(i,m) -1 + fs(i,m,j))*x(j,»
cons3..          sum({i,t,m},w(i)*q(i,m)*x(i,m,t)) =g= Qmin/card(i);
cons4..          sum({i,t,m},c(i,m)*x(i,m,t)) =l= B;

```

```

Model ProjectScheduling

```

```

/
objT
objC
objQ
cons1
cons2
cons3
cons4
/
;

```

```

Options

```

```

MIP = CPLEX
reslim =100
*maximum run time (sec.)
optcr = 0
;

```

```

***** Pay-off

```

```

Set Objective /Cost, Quality, Time/;

```

```

Parameter

```

```

PayMat(Objective,Objective)
;

```

```

Solve ProjectScheduling us MIP min time;

```

```

PayMat('time','time')= time.l;
PayMat('Cost','time')= Cost.l;
PayMat('Quality','time')= Quality.l;

```

```

Solve ProjectScheduling us MIP min cost;

```

```

PayMat('cost','cost')= cost.l;
PayMat('time','cost')= time.l;
PayMat('Quality','cost')= Quality.l;

```

OR

OptimYar

**Solve** ProjectScheduling us MIP max Quality;

```
PayMat('Quality','Quality')= Quality.l;  
PayMat('time','Quality')= time.l;  
PayMat('cost','Quality')= cost.l;
```

#### Parameter

```
MaxO(Objective)  
MinO(Objective)  
Rang(Objective)  
;
```

```
MaxO('cost') = smax(Objective, PayMat('cost', Objective));  
MinO('cost') = smin(Objective, PayMat('cost', Objective));
```

```
MaxO('time') = smax(Objective, PayMat('time', Objective));  
MinO('time') = smin(Objective, PayMat('time', Objective));
```

```
MaxO('Quality') = smax(Objective, PayMat('Quality', Objective));  
MinO('Quality') = smin(Objective, PayMat('Quality', Objective));
```

```
Rang(Objective)= MaxO(Objective) - MinO(Objective);
```

#### Display

```
PayMat  
MaxO  
MinO  
Rang  
;
```

```
***** Epsilon Constraint
```

#### Scalar

```
ET  
* MainO('time') <= ET <= MaxO('time')*  
EQ  
* MainO('Quality') <= EQ <= MaxO('Quality')*  
;
```

#### Equations

```
ecTime  
ecquality  
;  
ecTime.. time =l= ET ;  
ecquality .. quality =g= EQ ;
```

OR

OptimYar

```
Models    EC_ProjectScheduling
/
ProjectScheduling
ecTime
ecquality
/
;
```

```
For ( ET = MinO('time') to MaxO('time') by Rang('time')/5 ,
```

```
For ( EQ = MaxO('Quality') downto MinO('Quality') by Rang('Quality')/5 ,
```

```
Solve EC_ProjectScheduling us MIP min cost;
```

```
Display
```

```
"ET"
```

```
ET
```

```
"EQ"
```

```
EQ
```

```
x.1
```

```
Cost.1
```

```
Time.1
```

```
Quality.1
```

```
);
```

```
);
```



OptimYar