

## Assignment 6

Anuraag Vasal

11/22/2021

```
library(lpSolveAPI)
BIP<- read.lp("Assignment 6-1.lp")
solve(BIP)
```

```
## [1] 0
```

```
#Objective
```

```
get.objective(BIP)
```

```
## [1] 17
```

```
#Variables
```

```
get.variables(BIP)
```

```
## [1] 1 0 0 1 0 0 0 0 1 0 1 0
```

```
#Constraints
```

```
get.constraints(BIP)
```

```
## [1] 1 0 0 0 0 0 0 0 1
```

The longest path is 17.

\*Question 2(a)

```
Stocks <- make.lp(0,8)
lp.control(Stocks,sense="max")
```

```
## $anti.degen
```

```
## [1] "fixedvars" "stalling"
```

```
##
```

```
## $basis.crash
```

```
## [1] "none"
```

```
##
```

```
## $bb.depthlimit
```

```
## [1] -50
```

```
##
```

```
## $bb.floorfirst
```

```
## [1] "automatic"
```

```
##
```

```
## $bb.rule
```

```
## [1] "pseudononint" "greedy" "dynamic" "rcostfixing"
```

```
##
```

```
## $break.at.first
```

```

## [1] FALSE
##
## $break.at.value
## [1] 1e+30
##
## $epsilon
##      epsb      epsd      epsel      epsint  epsperturb  epspivot
##      1e-10      1e-09      1e-12      1e-07      1e-05      2e-07
##
## $improve
## [1] "dualfeas" "thetagap"
##
## $infinite
## [1] 1e+30
##
## $maxpivot
## [1] 250
##
## $mip.gap
## absolute relative
##      1e-11      1e-11
##
## $negrange
## [1] -1e+06
##
## $obj.in.basis
## [1] TRUE
##
## $pivoting
## [1] "devex"      "adaptive"
##
## $presolve
## [1] "none"
##
## $scalelimit
## [1] 5
##
## $scaling
## [1] "geometric"  "equilibrate" "integers"
##
## $sense
## [1] "maximize"
##
## $simplextype
## [1] "dual"      "primal"
##
## $timeout
## [1] 0
##

```

```

## $verbose
## [1] "neutral"

set.objfn(Stocks,c(4,6.5,5.9,5.4,5.15,10,8.4,6.25))
set.type(Stocks,c(1:8), type = "integer")
add.constraint(Stocks,c(40,50,80,60,45,60,30,25),"<=",2500000,indices =
c(1:8))
add.constraint(Stocks,1000,">=",0,indices = 1)
add.constraint(Stocks,1000,">=",0,indices = 2)
add.constraint(Stocks,1000,">=",0,indices = 3)
add.constraint(Stocks,1000,">=",0,indices = 4)
add.constraint(Stocks,1000,">=",0,indices = 5)
add.constraint(Stocks,1000,">=",0,indices = 6)
add.constraint(Stocks,1000,">=",0,indices = 7)
add.constraint(Stocks,1000,">=",0,indices = 8)
add.constraint(Stocks,40,">=",100000,indices = 1)
add.constraint(Stocks,50,">=",100000,indices = 2)
add.constraint(Stocks,80,">=",100000,indices = 3)
add.constraint(Stocks,60,">=",100000,indices = 4)
add.constraint(Stocks,45,">=",100000,indices = 5)
add.constraint(Stocks,60,">=",100000,indices = 6)
add.constraint(Stocks,30,">=",100000,indices = 7)
add.constraint(Stocks,25,">=",100000,indices = 8)
add.constraint(Stocks,c(40,50,80),"<=",1000000,indices = c(1,2,3))
add.constraint(Stocks,c(60,45,60),"<=",1000000,indices = c(4,5,6))
add.constraint(Stocks,c(30,25),"<=",1000000,indices = c(7,8))
solve(Stocks)

## [1] 0

#Objective
get.objective(Stocks)

## [1] 487145.2

#Variables
get.variables(Stocks)

## [1] 2500 6000 1250 1667 2223 13332 30000 4000

#Constraints
get.constraints(Stocks)

## [1] 2499975 2500000 6000000 1250000 1667000 2223000 13332000
300000000
## [9] 4000000 100000 300000 100000 100020 100035 799920
900000
## [17] 100000 500000 999975 1000000

```

\*Question 2(b)

```

Stocks1<-make.lp(0,8)
lp.control(Stocks1,sense="max")

## $anti.degen
## [1] "fixedvars" "stalling"
##
## $basis.crash
## [1] "none"
##
## $bb.depthlimit
## [1] -50
##
## $bb.floorfirst
## [1] "automatic"
##
## $bb.rule
## [1] "pseudononint" "greedy"          "dynamic"          "rcostfixing"
##
## $break.at.first
## [1] FALSE
##
## $break.at.value
## [1] 1e+30
##
## $epsilon
##      epsb      epsd      epsel      epsint  epsperturb  epspivot
##      1e-10      1e-09      1e-12      1e-07       1e-05       2e-07
##
## $improve
## [1] "dualfeas" "thetagap"
##
## $infinite
## [1] 1e+30
##
## $maxpivot
## [1] 250
##
## $mip.gap
## absolute relative
##      1e-11      1e-11
##
## $negrange
## [1] -1e+06
##
## $obj.in.basis
## [1] TRUE
##
## $pivoting
## [1] "devex"      "adaptive"
##

```

```

## $presolve
## [1] "none"
##
## $scalelimit
## [1] 5
##
## $scaling
## [1] "geometric" "equilibrate" "integers"
##
## $sense
## [1] "maximize"
##
## $simplextype
## [1] "dual" "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"

set.objfn(Stocks1,c(4,6.5,5.9,5.4,5.15,10,8.4,6.25))
add.constraint(Stocks1,c(40,50,80,60,45,60,30,25),"<=",2500000,indices =
c(1:8))
add.constraint(Stocks1,1000,">=",0,indices = 1)
add.constraint(Stocks1,1000,">=",0,indices = 2)
add.constraint(Stocks1,1000,">=",0,indices = 3)
add.constraint(Stocks1,1000,">=",0,indices = 4)
add.constraint(Stocks1,1000,">=",0,indices = 5)
add.constraint(Stocks1,1000,">=",0,indices = 6)
add.constraint(Stocks1,1000,">=",0,indices = 7)
add.constraint(Stocks1,1000,">=",0,indices = 8)
add.constraint(Stocks1,40,">=",100000,indices = 1)
add.constraint(Stocks1,50,">=",100000,indices = 2)
add.constraint(Stocks1,80,">=",100000,indices = 3)
add.constraint(Stocks1,60,">=",100000,indices = 4)
add.constraint(Stocks1,45,">=",100000,indices = 5)
add.constraint(Stocks1,60,">=",100000,indices = 6)
add.constraint(Stocks1,30,">=",100000,indices = 7)
add.constraint(Stocks1,25,">=",100000,indices = 8)
add.constraint(Stocks1,c(40,50,80),"<=",1000000,indices = c(1,2,3))
add.constraint(Stocks1,c(60,45,60),"<=",1000000,indices = c(4,5,6))
add.constraint(Stocks1,c(30,25),"<=",1000000,indices = c(7,8))
solve(Stocks1)

## [1] 0

#Objective
get.objective(Stocks1)

## [1] 487152.8

```

### *#Variables*

```
get.variables(Stocks1)
```

```
## [1] 2500.000 6000.000 1250.000 1666.667 2222.222 13333.333 30000.000
```

```
## [8] 4000.000
```

### *#Constraints*

```
get.constraints(Stocks1)
```

```
## [1] 2500000 2500000 6000000 1250000 1666667 2222222 13333333  
30000000
```

```
## [9] 4000000 100000 300000 100000 100000 100000 800000  
900000
```

```
## [17] 100000 500000 1000000 1000000
```