Assignment2

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com (http://rmarkdown.rstudio.com).

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
library(lpSolveAPI)
Weigelt_prod.lp <- make.lp(11,9)

#using function max

set.objfn(Weigelt_prod.lp, c(420,360,300,420,360,300,420,360,300))
lp.control(Weigelt_prod.lp, 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"
```

```
# available storage space

set.row(Weigelt_prod.lp, 1, c(20,15,12), indices = c(1,2,3))
set.row (Weigelt_prod.lp, 2, c(20,15,12), indices = c(4,5,6))
set.row(Weigelt_prod.lp,3, c(20,15,12), indices = c(7,8,9))

# Plants excess capacity

set.row(Weigelt_prod.lp, 4, c(1,1,1), indices = c(1,2,3))
set.row(Weigelt_prod.lp, 5, c(1,1,1), indices = c(4,5,6))
set.row(Weigelt_prod.lp,6, c(1,1,1), indices = c(7,8,9))

## sales forecast

set.row(Weigelt_prod.lp, 7, c(1,1,1), indices = c(1,4,7))
set.row(Weigelt_prod.lp, 8, c(1,1,1), indices = c(2,5,8))
set.row(Weigelt_prod.lp, 9, c(1,1,1), indices = c(3,6,9))

#Percentage

set.row(Weigelt_prod.lp, 10, c(0.0013, 0.0013, 0.0013,-0.0011,-0.0011,-0.0011), indices
 = c(1,2,3,4,5,6))
set.row(Weigelt_prod.lp, 11, c(0.0013,0.0013,0.0013,-0.0022,-0.0022,-0.0022), indices =
 c(1,2,3,7,8,9))

rhs <- c(13000,12000,5000,750, 900, 450, 900, 1200, 750,0,0)
set.rhs (Weigelt_prod.lp, rhs)
set.constr.type(Weigelt_prod.lp, c("<=","<=","<=","<=","<=","<=","<=","<=","<=","=","="
))

#names

vname <- c("PL1","PM1","PS1","PL2", "PM2","PS2", "PL3","PM3","PS3")
constraint_name <- c("St1", "St2","St3", "Ex1","Ex2","Ex3","S1", "S2","S3", "Percentage
A", "PercentageB")

Weigelt_prod.lp
```

```
## Model name:
##   a linear program with 9 decision variables and 11 constraints
```

```
solve(Weigelt_prod.lp)
```

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

```
get.objective(Weigelt_prod.lp)
```

```
## [1] 697923.1
```

```
solution <- data.frame(vname, get.variables (Weigelt_prod.lp))
colnames(solution) <- c("value1", "value2")
solution
```

```
##   value1   value2
## 1    PL1 484.6154
## 2    PM1 220.5128
## 3    PS1   0.0000
## 4    PL2   0.0000
## 5    PM2 666.6667
## 6    PS2 166.6667
## 7    PL3   0.0000
## 8    PM3   0.0000
## 9    PS3 416.6667
```