

## Assignment 2

1. (Computer Center Staffing) You are the Director of the Computer Center for Gaillard College and responsible for scheduling the staffing of the center. It is open from 8 am until midnight. You have monitored the usage of the center at various times of the day and determined that the following numbers of computer consultants are required.

Time of day	Minimum number of consultants required to be on duty
8 am–noon	4
Noon–4 pm	8
4 pm–8 pm	10
8 pm–midnight	6

Two types of computer consultants can be hired: full-time and part-time. The full-time consultants work for eight consecutive hours in any of the following shifts: morning (8 am –4 pm), afternoon (noon –8 pm), and evening (4 pm –midnight). Full-time consultants are paid \$14 per hour.

Part-time consultants can be hired to work any of the four shifts listed in the table. Part-time consultants are paid \$12 per hour. An additional requirement is that during every period, at least one full-time consultant must be on duty for every part-time consultant on duty.

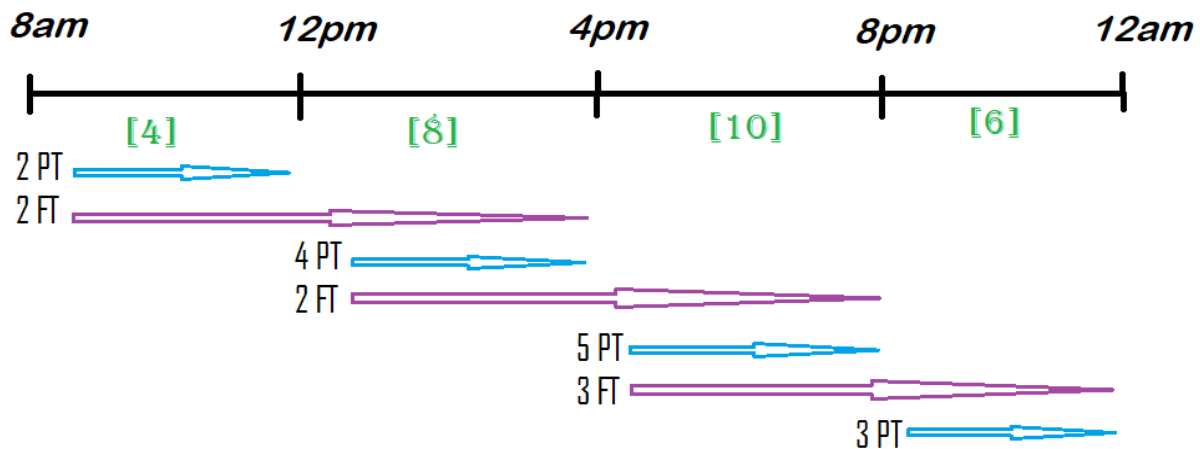
- a) Determine a minimum-cost staffing plan for the center. In your solution, how many consultants will be paid to work full time and how many will be paid to work part time? What is the minimum cost?

Cost factor for each type of consultant: Full-time = \$14/hr, while Part-time = \$12/hr. To minimize the cost; Part-time consultants should be hired to the entire assignment. Because this will reduce the cost factor by \$2/hr. But every part time can't work independently and requires a Full-time consultant's supervision over their work. Hence the best approach would be maximizing the count of Part-time consultants in operations.

$$\begin{aligned} Z &= (14) (8) x + (12) (4) y && [x: \text{full-time consultants \& } y: \text{Half-time consultants}] \\ 8x + 4y &= 16 && \{\text{Condition 1}\} \\ x &\geq y && \{\text{Condition 2}\} \end{aligned}$$

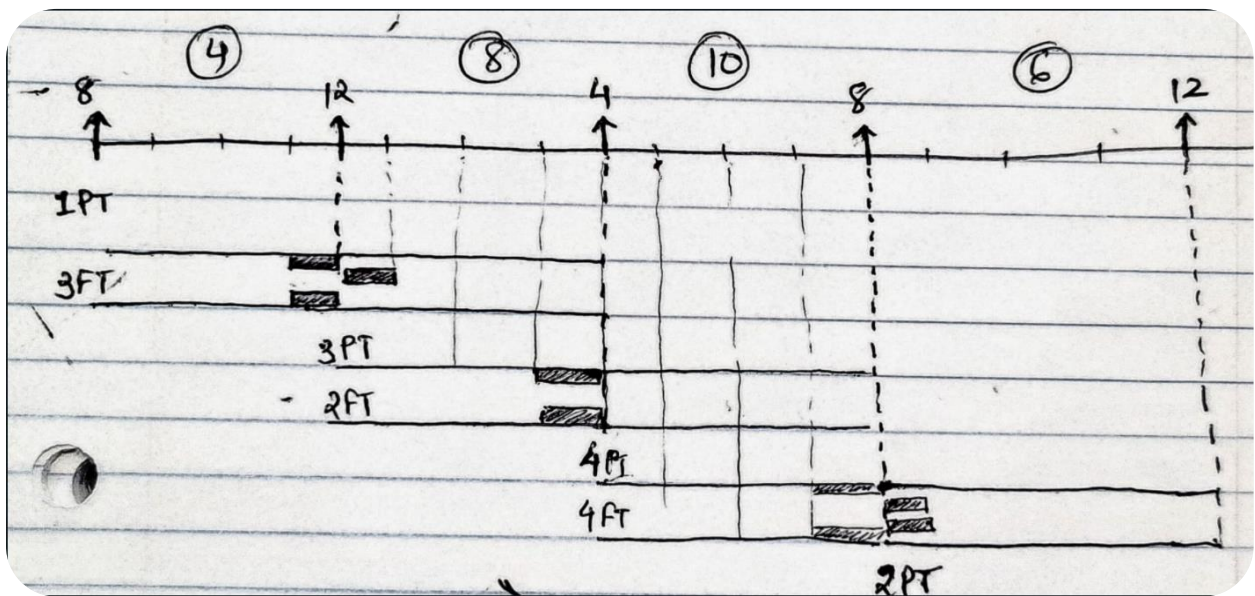
The best combination is shown as below; Objective is to maximize usage of half-time consultants.

14 Part-time consultants and 7 full-time consultants make the best combination as shown in the figure to minimize cost.



$$[ 14 \times 4(\text{hrs}) \times 12 (\$/\text{hr}) ] + [ 7 \times 8(\text{hrs}) \times 14(\$/\text{hr}) ] = 1,465 \$$$

- b) After thinking about this problem for a while, you have decided to recognize meal breaks explicitly in the scheduling of full-time consultants. Full-time consultants are entitled to a one-hour lunch break during their eight-hour shift. In addition, employment rules specify that the lunch break can start after three hours of work or after four hours of work, but those are the only alternatives. Part-time consultants do not receive a meal break. Under these conditions, find a minimum-cost staffing plan. What is the minimum cost?



$$Z = (14) (8) x + (12) (4) y \quad (\text{Objective Function})$$

$$8x + 4y = 16 \quad \{\text{Condition 1}\}$$

$$x \geq y \quad \{\text{Condition 2}\}$$

$$x - 2 \geq y \quad \{\text{Condition 3}\}$$

In addition to the solution of the first part we will add a condition circumscribing the condition where full-time consultants are entitled to take breaks after three or four hours from their start of the shift. Condition 3; number of full-time consultants actively working during break hours should be greater than or equal to part time consultants.

The best combination is shown as below:

10 Part-time consultants and 9 full-time consultants make the best combination as shown in the figure to minimize cost.

$$[ 10 \times 4(\text{hrs}) \times 12 (\$/\text{hr}) ] + [ 9 \times 8(\text{hrs}) \times 14(\$/\text{hr}) ] = 1,488 \$$$

## 2. Consider the problem from the previous assignment.

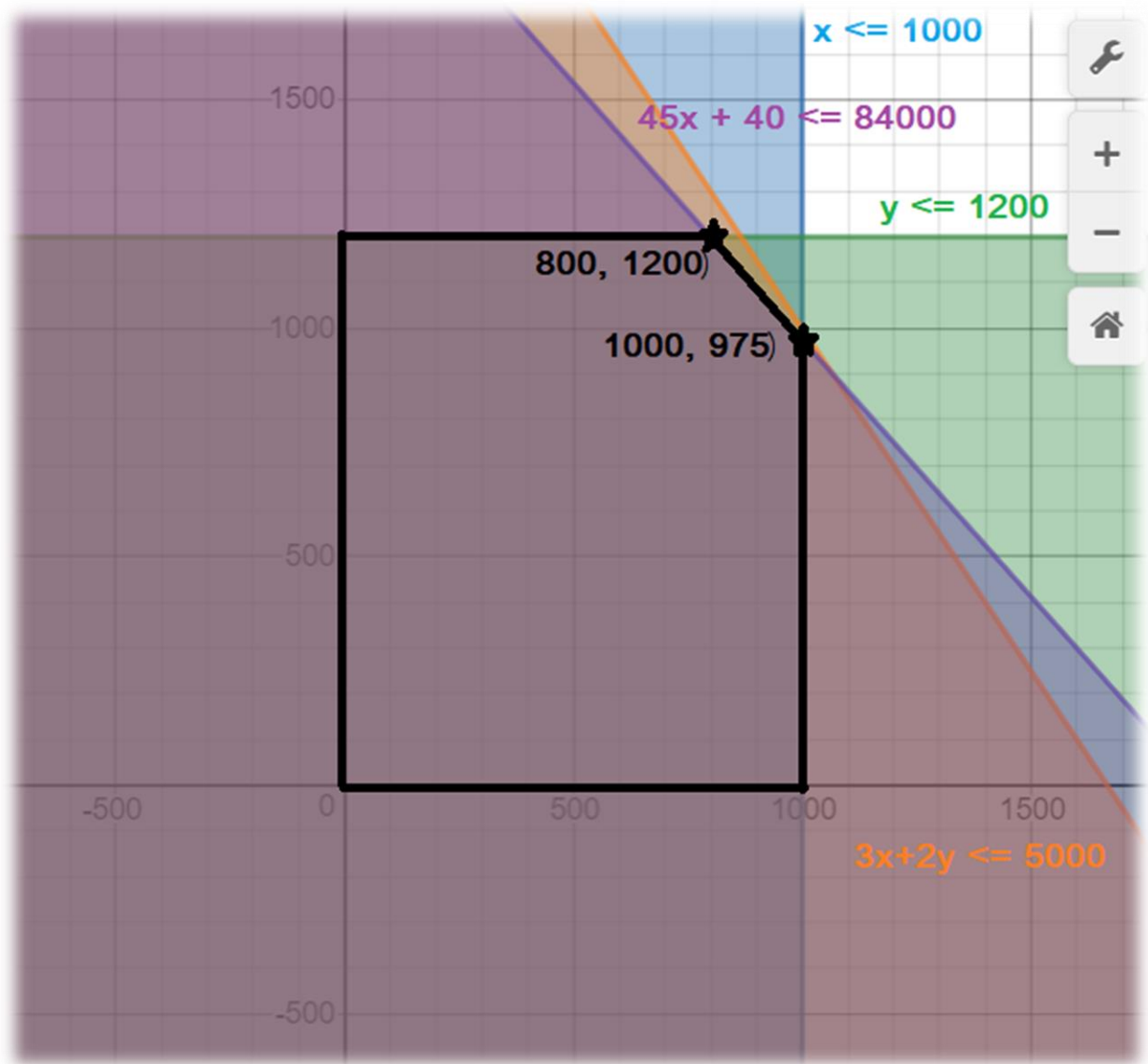
**Back Savers** is a company that produces backpacks primarily for students. They are considering offering some combination of two different models—the Collegiate and the Mini. Both are made from the same rip-resistant nylon fabric. Back Savers has a long-term contract with a supplier of the nylon and receives a 5000 square-foot shipment of the material each week. Each Collegiate requires 3 square feet while each Mini requires 2 square feet. The sales forecasts indicate that at most 1000 Collegiates and 1200 Minis can be sold per week. Each Collegiate requires 45 minutes of labor to produce and generates a unit profit of \$32. Each Mini requires 40 minutes of labor and generates a unit profit of \$24. Back Savers has 35 laborers that each provides 40 hours of labor per week. Management wishes to know what quantity of each type of backpack to produce per week. Solve this problem graphically.

The company must produce X collegiate bags and Y mini backpacks per week to maximize the profit.

$$Z = \$(32x + 24y) \quad (\text{Objective function})$$

$$3x + 2y \leq 5000 \quad (\text{Practical Condition})$$

$$x \leq 1000, y \leq 1200$$



3. (Weigelt Production) The Weigelt Corporation has three branch plants with excess production capacity. Fortunately, the corporation has a new product ready to begin production, and all three plants have this capability, so some of the excess capacity can be used in this way. This product can be made in three sizes--large, medium, and small--that yield a net unit profit of \$420, \$360, and \$300, respectively. Plants 1, 2, and 3 have the excess capacity to produce 750, 900, and 450 units per day of this product, respectively, regardless of the size or combination of sizes involved. The amount of available in-process storage space also imposes a limitation on the production rates of the new product. Plants 1, 2, and 3 have 13,000, 12,000, and 5,000 square feet, respectively, of in-process storage space available for a day's production of this product. Each unit of the large, medium, and small sizes produced per day requires 20, 15, and 12 square feet, respectively. Sales forecasts indicate that if available, 900, 1,200, and 750 units of the

large, medium, and small sizes, respectively, would be sold per day. At each plant, some employees will need to be laid off unless most of the plant's excess production capacity can be used to produce the new product. To avoid layoffs, if possible, management has decided that the plants should use the same percentage of their excess capacity to produce the new product. Management wishes to know how much of each of the sizes should be produced by each of the plants to maximize profit.

**a. Define the decision variables.**

Let  $x_{ij}$  be the number of units of size  $j$  ( $j$ =Large, Medium, Small) product produced in plant  $i$  ( $i$ =1,2,3)

hence, the decision variables are  $x_{1l}, x_{1m}, x_{1s}, x_{2l}, x_{2m}, x_{2s}, x_{3l}, x_{3m}$  and  $x_{3s}$

**b. Formulate a linear programming model for this problem.**

Answer referenced in QMM\_Assignment2.lp file

```
/* Objective function */
max: +420 P1Large +360 P1Medium +300 P1Small +420 P2Large +360 P2Medium +300 P2Small +420 P3Large +360 P3Medium
+300 P3Small;

/* Constraints */
CapCon1: +P1Large +P1Medium +P1Small <= 750;
CapCon2: +P2Large +P2Medium +P2Small <= 900;
CapCon3: +P3Large +P3Medium +P3Small <= 450;
StoCon1: +20 P1Large +15 P1Medium +12 P1Small <= 13000;
StoCon2: +20 P2Large +15 P2Medium +12 P2Small <= 12000;
StoCon3: +20 P3Large +15 P3Medium +12 P3Small <= 5000;
SalCon1: +P1Large +P1Medium +P1Small <= 900;
SalCon2: +P2Large +P2Medium +P2Small <= 1200;
SalCon3: +P3Large +P3Medium +P3Small <= 750;
%C1: +6 P1Large +6 P1Medium +6 P1Small -5 P2Large -5 P2Medium -5 P2Small = 0;
%C2: +3 P1Large +3 P1Medium +3 P1Small -5 P3Large -5 P3Medium -5 P3Small = 0;
```

**c. Solve the problem using lpSolve, or any other equivalent library in R.**

```
library(lpSolveAPI)

## Warning: package 'lpSolveAPI' was built under R version 4.0.3

# setwd("~/R_KSU/Quant/Assignment2")

lprec <- make.lp(0,9)
lprec

## Model name:
## a linear program with 9 decision variables and 0 constraints

set.objfn(lprec, c(420,360, 300, 420, 360, 300, 420, 360, 300))
lp.control(lprec, sense='max')

## $anti.degen
## [1] "none"
##
## $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"

# Add the constraints
add.constraint(lprec, c(1, 1, 1, 0, 0, 0, 0, 0, 0), "<=", 750)
add.constraint(lprec, c(0, 0, 0, 1, 1, 1, 0, 0, 0), "<=", 900)
add.constraint(lprec, c(0, 0, 0, 0, 0, 0, 1, 1, 1), "<=", 450)
add.constraint(lprec, c(20, 15, 12, 0, 0, 0, 0, 0, 0), "<=", 13000)
add.constraint(lprec, c(0, 0, 0, 20, 15, 12, 0, 0, 0), "<=", 12000)
add.constraint(lprec, c(0, 0, 0, 0, 0, 0, 20, 15, 12), "<=", 5000)
add.constraint(lprec, c(1, 1, 1, 0, 0, 0, 0, 0, 0), "<=", 900)
add.constraint(lprec, c(0, 0, 0, 1, 1, 1, 0, 0, 0), "<=", 1200)
add.constraint(lprec, c(0, 0, 0, 0, 0, 0, 1, 1, 1), "<=", 750)
add.constraint(lprec, c(6, 6, 6, -5, -5, -5, 0, 0, 0), "=", 0)
add.constraint(lprec, c( 3, 3, 3, 0, 0, 0, -5, -5, -5), "=", 0)

RowNames <- c("CapCon1", "CapCon2", "CapCon3", "StoCon1", "StoCon2", "StoCon3",
              "SalCon1", "SalCon2", "SalCon3", "%C1", "%C2")
ColNames <- c("P1Large", "P1Medium", "P1Small", "P2Large", "P2Medium", "P2Small",
              "P3Large", "P3Medium", "P3Small")
dimnames(lprec) <- list(RowNames, ColNames)

lprec

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

write.lp(lprec, filename = "QMM_Assignment2.lp", type = "lp")
solve(lprec)

## [1] 0

get.objective(lprec)

## [1] 696000

get.variables(lprec)

## [1] 516.6667 177.7778    0.0000    0.0000 666.6667 166.6667    0.0000    0.0000
0
## [9] 416.6667

```