R Notebook

```
library(DiagrammeR)
library(webshot)
```

Product Line Balancing Problem

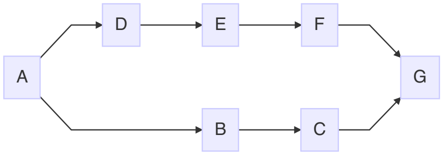
```
dt <- data.frame("task" = c("A", "B", "C", "D", "E", "F", "G", "G"),
                 "time" = c(60, 12, 35, 55, 10, 50, 5, 5),
                 "predecessor" = c(NA, "A", "B", "A", "D", "E", "F", "C"))
forecastedDemand <- 40
```

1 Precedence Relationship Network

```
writeLines(paste("Total time:",
                 totalTime <- sum(unique(dt[, c("task", "time")])[, "time"])))
```

```
## Total time: 227
```

```
mermaid(c("graph LR;",
          apply(dt[rowSums(is.na(dt)) == 0,], 1, function(x) {
            paste0(x["predecessor"], " --> ", x["task"], ";")
            })
          ))
```



2 Determine the Output Rate per hour (day)

```
writeLines(paste("The maximum output per hour is:",
                 maxOutput <- 3600 / totalTime))
```

```
## The maximum output per hour is: 15.8590308370044
```

```
writeLines(paste("The maximum output per 8 hour workday is:", maxOutput * 8))
```

```
## The maximum output per 8 hour workday is: 126.872246696035
```

```
if(forecastedDemand > maxOutput) {
  writeLines(paste("Forecasted demand is greater than maximum output so an",
                   "assembly line with work stations is a possibility"))
} else {
  writeLines(paste("Forecasted demand is less than maximum output so an",
                   "assembly line with work stations is not a possibility"))
}
```

```
## Forecasted demand is greater than maximum output so an assembly line with work stations is a possibil
```

3 Determine the takt time and maximum bottleneck output

```r
writeLines(paste("The takt time is:", taktTime <- 3600 / forecastedDemand))
```

```
## The takt time is: 90
```

```r
writeLines(paste("The maximum bottleneck output is:",
                 bottleneckOutput <- 3600 / max(dt$time)))
```

```
## The maximum bottleneck output is: 60
```

4 Compute the theoretical minimum workstations

```r
writeLines(paste("The theoretical minimum workstations is:",
                 minimumStations <- totalTime / taktTime))
```

```
## The theoretical minimum workstations is: 2.52222222222222
```

```r
if(minimumStations %% 1 != 0) {
  writeLines(paste("However, because you cannot have a partial workstation,",
                   "the real number is:",
                   minimumStations <- ceiling(minimumStations)))
}
```

```
## However, because you cannot have a partial workstation, the real number is: 3
```

5 Assign tasks to workstations

Work Station	Eligible Tasks	Task Selected	Time	Idle
1	AB	AB	72	18
2	CD	CD	90	0
3	EFG	EFG	65	25

```r
stations <- data.frame("station" = 1:ceiling(minimumStations),
                       "eligible" = c("AB", "CD", "EFG"),
                       "selected" = c("AB", "CD", "EFG"),
                       "time" = c(72, 90, 65),
                       "idle" = c(18, 0, 25))
```

6 Compute Efficiency and balance delay

```r
writeLines(paste0("The efficiency is: ",
                  efficiency <- totalTime /
                    (taktTime * minimumStations) * 100, "%"))
```

```
## The efficiency is: 84.0740740740741%
```

```r
writeLines(paste0("The balance delay is: ", 100 - efficiency, "%"))
```

```
## The balance delay is: 15.9259259259259%
```