Define an algorithm in Matlab for the following scheduling problem.

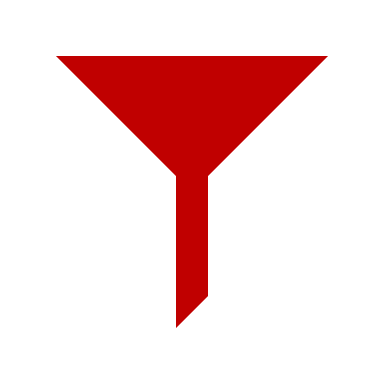
Schedule the following *n* jobs, where *pj* is the processing time on machine i. The goal is to minimize Cmax.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Job | J1 | J2 | J3 | J4 | J5 | J6 | J7 | J8 | J9 | J10 |
| *P1* | 5 | 3 | 6 | 8 | 4 | 12 | 12 | 5 | 3 | 2 |
| *P2* | 12 | 6 | 1 | 5 | 6 | 15 | 3 | 2 | 8 | 8 |
| *P3* | 1 | 20 | 2 | 5 | 7 | 11 | 12 | 2 | 5 | 4 |
| *P4* | 1 | 1 | 1 | 15 | 6 | 2 | 1 | 4 | 4 | 3 |
| *P5* | 2 | 6 | 2 | 11 | 5 | 3 | 2 | 7 | 8 | 3 |

Verify that the solution that has been obtained is optimal or not comparing with the solution obtained in a mathematical programming problem defined in Excel (or other spreadsheet tool with optimization module) or Lingo or Cplex or Matlab.

PROJECT SOLUTION

**PART 1) Problem:** we have five machines (Mn) and ten jobs (Jn) to schedule into the five machines. There is and issue in the system presented because the jobs have 2 possible routes that converge into only one machine (M4) with the risk of a bottleneck.

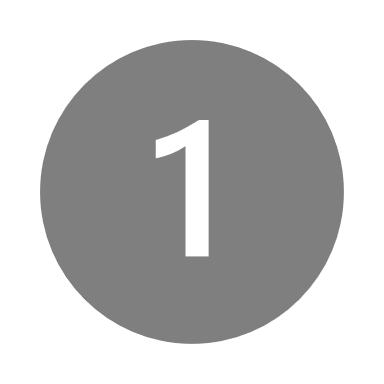


Bottleneck risk

So the idea is to have a smart way to define the order and how to route the jobs to avoid the bottleneck into the machine 4. Our proposal is to consider machine 2 and machine 3 as a unique machine exploiting the parallelism between the two machines.

After deciding how to manage the machines, we sorted the jobs in ascending order by duration of execution of the jobs on machines 1 to 4/5.

At this point we divided them into two groups: **Way2** for the jobs with short time compared to the other group and **Way3** with the more slow time jobs.





Way2



Way3

