



INF2220: algorithms and data structures

Mandatory assignment 2

Issued: 28. 09. 2016

Due:

1 Optional task: Limited manpower

For those who want to, they can include functionality to deal with limited manpower. When the manpower is limited there are some new factors to running a project. Like with unlimited manpower a task may be started once all its predecessors have been completed, and in addition there must be enough free manpower so work on it. When it comes to realizability no task can demand more manpower than is totally available, if so that task cannot be completed.

Whereas the problem with limited manpower is easily solvable using simple algorithms you have been taught in this course, unlimited manpower is a much harder problem. It is actually (computationally) hard to calculate an optimal schedule.

The task therefore is not even to attempt that, but make a “common sense” approach (one could say, a *heuristic* approach) which generalizes the one for unlimited manpowers: simply start at each point as many tasks as possible and this way at least avoid “idling time” where tasks *could* be started but are not yet handled.

This can be done with a greedy method simply starting a task if there is enough manpower ignoring the possibility that it would have been more effective to start another task instead.

Implementation

If you choose to deal with limited manpower you should solve the problem with unlimited manpower first and generalize your solution to support limited manpower. The result of running your program with sufficient limited manpower (you have given the program enough manpower so it will not delay the project) should be the same as for the algorithm handling unlimited manpower.