

Advanced Programming (I00032) 2018

Getting the job done

Assignment 6

Goal

In this assignment you will make a larger program using the `iTask` DSL. This should give you more experience in the use of this DSL. Please pay some attention to the rationale behind the design of the language, what works convenient, and the DSL problems you encounter.

Part of your final mark

To reward your efforts in making this larger assignment it can contribute to your final mark of this course. When your mark for the exam is at least 5, the final mark can be improved by averaging with this exercise. The weight of this exercise will be 25%. The marks will only be averaged if this improves your result. This implies that you will receive a normal mark for this assignment between 1 and 10.

1 Getting the job done

In this assignment you will make a simple system to create and execute *jobs*. There is one central store with jobs. Each jobs has at least a name and a set of skills required to execute that job. You can use a datatype like:

```
:: Job =  
  { jobName      :: String  
    , skillsNeeded :: [Skill]  
  }
```

Add any field to this record that suits your needs, e.g. a unique number if that is required. Choose a convenient implementation for `Skill`, e.g. a simple `String` or some algebraic data type.

2 Workers

Any number of workers can participate in this program. Such a worker has at least name and a set of skills. Each worker sees all jobs that require only skills that she has. Each worker can perform the following actions:

- Create a new job that is added to the list.
- Change her set of skills.

- Select one of the existing jobs. For the selected job the worker can:
 - Cancel the job execution, nothing will be changed.
 - Execute the job. The completed job will disappear from the list of jobs to be done.
 - Split the job in a list of sub-jobs. The original job will be blocked until all sub-jobs are done.

Jobs that are completed or blocked cannot be selected for execution.

2.1 Hints

By construction this assignment leaves much freedom of design, many things are only partly specified. You are free to make your own decisions and design choices. List the main choices by a brief comment in your code.

Deadline

You have additional time to make this exercise since there is two weeks break for exams. There is no exam for this course in those weeks. The deadline for this exercise is November 5, 2018, 10:30h (just before the next lecture).

Hand in the code, and a very brief description of the main decisions made in the implementation and some screen shots of your system. This should make it easier to mark your work. It is recommended to make this assignment with one partner.

Please contact us when you encounter problems.