

BLG 312E – Computer Operating Systems

Homework 3

Submission Deadline: 12.05.2017, 23:55

(Late Submission Deadline: 13.05.2017, 23:55

- You are expected to work individually on all exams and homeworks. All forms of collaboration are discouraged and will be treated as cheating. This includes actions such as, but not limited to, collaborating with another student, submitting the work of others as one's own (even if in part and even with modifications) and copy/pasting from other resources (including Internet resources) even when attributed. Serious offenses will be reported to the administration for disciplinary measures. All parties involved in the act will be held equally responsible.
- You have to achieve at least 20 out of 100 points on a homework for its submission to be accepted. Homeworks with lower grades will NOT be considered as submitted. Submitting parts of the codes provided in class will NOT be sufficient to achieve a grade of 20.
- Submission of the 3rd homework is compulsory for being allowed to take the final exam.
- Late submissions will be allowed for only 24 hours after the submission deadline. Regardless of the time of the late submission and its reason, **ALL** late submissions will be graded as 50% of the original grade the homework deserves. (Please note that in the case of late submissions, for a homework to be considered as submitted, its original (unreduced) grade should be at least 40).

What to submit: You should submit your source file(s) via the Ninova system. (No additional report file is required; however, it is expected that you include comments in your source file.)

Note: Please only use English letters in your submission files, even for your names.

Program: Write and test a C program that implements the described behavior below:
In this homework, there will be three processes with multiple threads.

Description: In a company hiring new employees is done in two stages. In the first stage, the “human resources (HR) department” deals with the new applicants. The second stage is handled by the “trial department”.

In the HR department, there is one receptionist and three interviewers. When a new applicant arrives at the department, the receptionist registers him/her and assigns his/her application status as 0. After this step, the applicant goes into the waiting room to wait for an interviewer to become available. Whenever one of the interviewers becomes available, he/she calls the first applicant in line and interviews him/her. While the interviewers interview the applicants, the

receptionist continues registering incoming new applicants. (Note: Assume that the waiting room has infinite capacity.)

At the end of the interview, the applicant is either accepted to go into the trial period or is rejected. If an applicant is accepted to go into the trial period, the interviewer marks his/her status as 2; if he/she is rejected, the interviewer marks his/her status as 1. At this stage, rejected applicants leave the system. Applicants which have been accepted into the trial period go to the waiting room of the trial department to wait for a trainer to become available.

In the trial department, there are four trainers. Whenever one of the trainers becomes available, he/she calls the first applicant in line and starts training him/her. At the end of the trial period, the applicant is either fully accepted into the position he/she applied for or is rejected. If an applicant is fully accepted, the trainer marks his/her status as 4; if he/she is rejected, the trainer marks his/her status as 3. At the end of this stage, both rejected applicants and fully accepted applicants leave the system.

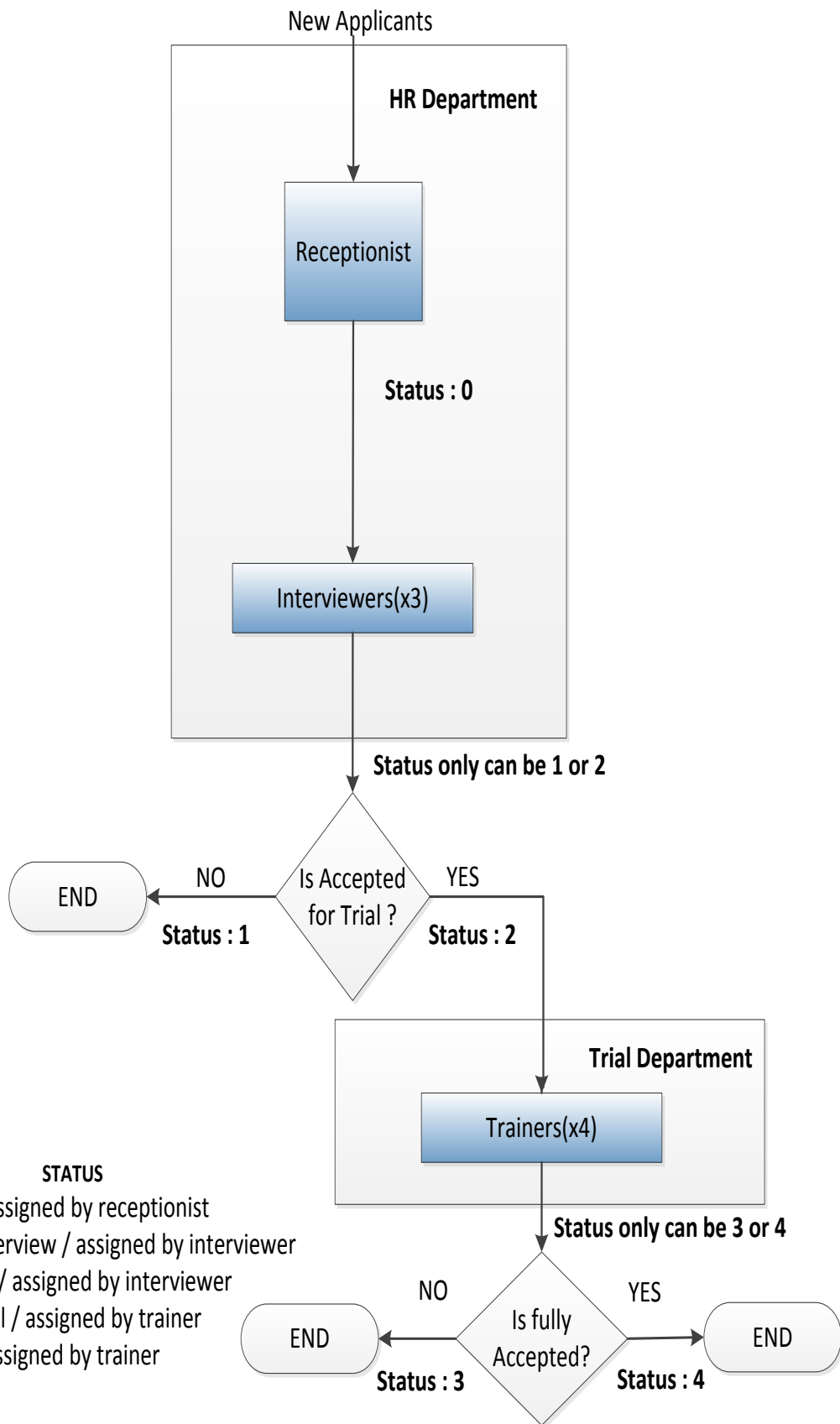
You are required to implement the following models. Please note that any other model will NOT get any points.

- Each applicant is a process with no threads.
- The HR department is a process with the interviewers and the receptionist as threads of this process. There are three interviewers and one receptionist.
- The trial department is a process with the trainers as threads of this process. There are four trainers.
- The *status* of each applicant is a shared integer variable which can be accessed (read) by the applicant and modified by the receptionist, the interviewers and the trainers.

Status information:

- “0” means that it is a new applicant who has been registered by the receptionist. This status is assigned by the receptionist when registration is completed.
- “1” means that the applicant has been rejected after the interview. This status is assigned by an interviewer.
- “2” means that the applicant has been accepted to go into the trial period after the interview. This status is assigned by an interviewer.
- “3” means that the applicant has been rejected after the trial period. This status is assigned by a trainer.
- “4” means that the applicant has been fully accepted for the position he/she applied for. This status is assigned by a trainer.

You can examine the flowchart provided below to get detailed information about the steps of this system.



To simulate the working of this system, inputs will be provided by way of an input text file. The file will contain the following information:

- n_r seconds: the time the receptionist needs to register each applicant (this time is fixed).
- Interviewing an applicant takes n_i minutes for an interviewer and n_i depends on the job position the applicant is applying for. To simulate applicants with different types of position applications, an interviewer thread should read the time it will take to interview the current applicant from the input file.
- Training an applicant in the trial department takes n_t minutes for a trainer n_t depends on the job position the applicant is applying for. To simulate applicants with different types of position applications, a trainer thread should read the time it will take to train the current applicant from the input file.
- The information whether the applicant is accepted or rejected at each stage is also given in the input file by the entry for each applicant.

Please note:

- The applicants do NOT have access to the input text file. Each applicant can only access his/her OWN status variable to see whether he/she has been accepted or rejected at each stage.
- The receptionist, interviewers and trainers can ONLY read the input file. They cannot make any changes on the file. They have to write the status of each applicant in the corresponding status variable of that applicant.
- To simulate the waiting times during the registration, interview and trial stages, you can use the “sleep” command.
- Please check that your program correctly removes all allocated resources (e.g. shared memory locations, semaphores, and any others you have used).
- You are required to use the following libraries:
 - semaphore.h or sys/sem.h for semaphore operations
 - pthread.h for threads
 - shm.h for shared memory operations

input file format: Input files for testing your program must be in the format given below where the first line holds the n_r variable to register the applicants. After the first line, each line contains the n_i variable for interviewing, n_t variable for training, i_t (0: rejected or 1:accepted) variable for training, i_j (0:rejected or 1:accepted) for full job for each applicant. Input file must be in “.txt” file format. Assume that there are 3 applicants with following information. For example, the first line holds 15 seconds which is the registration time for the receptionist (In the second homework this variable was taken from command line). Next line

holds, interview time is 2 minutes, training time is 5 minutes, 1(accepted) for training and 1(accepted) for full job for the first applicant. Second applicant (third line) is accepted for training but not for getting a job. (see its 4th column is 0.). Third applicant(forth line) is rejected after interview and can not be in the system and trained longer.

Input.txt

```
15
2 5 1 1
3 2 1 0
4 3 0 0
..
```

Test: Your program will be tested in the form:

./program

Please preserve the order and meaning of the program arguments. Please test your program with different input files and make sure to achieve expected results.

Output format: Your program must print the events on the screen in their order of occurrence in the format given below.

```
Registration time is: 15 seconds
Applicant 1 applied to the receptionist (status of applicant 1: 0)
Applicant 2 applied to the receptionist (status of applicant 2: 0)
Applicant 3 applied to the receptionist (status of applicant 3: 0)
Applicant 1's registration is done (status of applicant 1: 0)
Interviewer 1 started interview with Applicant 1 (status of applicant 1: 0)
Applicant 2's registration is done (status of applicant 2: 0)
Interviewer 3 started interview with Applicant 2 (status of applicant 2: 0)
Applicant 3's registration is done (status of applicant 3: 0)
Registration process is done.
Interviewer 2 started interview with Applicant 3 (status of applicant 3: 0)
Interviewer 1 finished interview with Applicant 1 (status of applicant 1: 2)
Trainer 2 started training with Applicant 1 (status of applicant 1: 2)
Interviewer 3 finished interview with Applicant 2 (status of applicant 2: 2)
Trainer 4 started training with Applicant 2 (status of applicant 2: 2)
Interviewer 2 finished interview with Applicant 3 (status of applicant 3: 1)
Interview process is done.
```

Trainer 4 finished training with Applicant 2 (status of applicant 2: 3)
Trainer 2 finished training with Applicant 1 (status of applicant 1: 4)
Training process is done.