Students in the library

1 Students in the library

Today's students assignment consists in reading some books that can be found in the school library. There are `nstudents` students and each student has a different list of books to read which is filled randomly. All lists contains the same number of books `nbooks` out of 50; this actually means that multiple students might have to read the same book but in the library there is only one copy of each book. In the library there are `ndesks` desks and, in order to read book, a student can use **any** desk; the problem is that the desks are very small and **only one person can use a desk** at any time.

2 Package content

In the `library` directory you will find the following files:

- `main.c`: this file contains the main program where each student gets her/his list of books by calling the `get_my_books_list` function. This list contains `nbooks` books identified by an integer value. Then the student goes through the list and, for each element of the list:

  1. tries to get the book if nobody else is reading it, otherwise she/he waits
  2. tries to get one free desk if any, otherwise she/he waits
  3. reads the book

  Only this file has to be modified for this exercise.

- `aux.c, aux.h`: these two files contain auxiliary routines and **must not be modified**.

The code can be compiled with the `make` command: just type `make` inside the `library` directory; this will generate a `main` program that can be run like this:

```
$ ./main nbooks ndesks nstudents
```

where `nbooks` is the number of books per student (the same for all students), `ndesks` the number of desks in the library and `nstudents` the number of students.

3 Assignment

-  parallelize the provided code respecting these constraints:

  1. Each thread must correspond to a student.
  2. Only one student can read a given book at some time; therefore if another student wants to read the same book, she/he has to wait.
  3. Only one student can use a desk at some time but each student can use any of the available desks.

4 Hints

You can use locks to prevent multiple students to read the same book and use the same desk at the same time.