EOPSY Lab5 – Synchronization, the Barber problem

Jakub Prochorec

The described problem is based on an analogy to the barber shop. In this exercise, there are three types of barbers – serving only women(N1), only men(N2) and serving both of them(N3). The two types of customers are women and men. There is waiting room with M chairs, and a cutting room. Clients wait in cutting room, but if they enter waiting room and there are no free chairs left, they leave the barber shop. Barbers check waiting room to see if there are clients that they can serve. If they are, they take them and cut hair for a random time. If not, they sleep for random time.

My solution uses semaphores, process forking and shared memory. For managing semaphores I’m using routines up, down, initSem. Up means semaphore is freed, down means it is locked, and initSem is used for initialization of semaphore. For shared memory implementation I’m using functions shmget, semget, shmat. I allocate single shared memory segment and then adjust pointers for different variables that I use(\*malewaiting, \*femalewaiting,\*waiting for tracking the number of customers waiting for haircut, and \*freefemalebarbers, \*freemalebarbers, \*freebothbarbers for tracking the numbers of free barbers). Then I’m forking 3 times for each type of the barber.

The program prints out messages from barbers and clients – barbers when they start cutting hair, and clients when they enter the shop(either sitting on a chair if there is one, or leaving the shop if all chairs in the waiting room are already taken). Functions srand and rand are used to generate random numbers. Those random numbers are used either to make some random time that it takes for barber to cut hair or sleep, or to decide if the person entering shop is male or female(in an evenly distributed way). Some extra details regarding the meaning of lines of the code are placed as comments within the program textfile itself. For the matter of clearance and honesty, there is one detail that I have made slightly different compared to the task description – in my solution, the barbers check the waiting room for customers, but the customers themselves do not enter cutting room without being invited – it is a kind gesture from customers not to enter the cutting room to disturb barbers or rudely wake them up in case they decided to take a quick nap having not noticed any clients. Overall, after all the setup and initialization there are 3 forks running – corresponding to male, female and both barbers. The variables N1,N2,N3 and M are adjustable at the top of the code section within the definitions. The program was tested for multiple combinations of these variables and produces reasonable output.