

IPC - pamięć wspólna, semaforey

Przydatne funkcje:

System V:

`<sys/shm.h>` `<sys/ipc.h>` - `shmget`, `shmctl`, `shmat`, `shmdt`

POSIX:

`<sys/mman.h>` - `shm_open`, `shm_close`, `shm_unlink`, `mmap`, `munmap`

Zadanie

Wykorzystując semaforey i pamięć wspólną z IPC Systemu V oraz standardu POSIX napisz program symulujący działanie salonu fryzjerskiego:

W salonie dostępnych jest:

M fryzjerów, N foteli (przy czym $M \geq N$), oraz poczekalnia z P miejscami.

Każdy fryzjer albo oczekuje na klienta (śpi), albo obsługuje klienta o ile taki się zgłosił bądź oczekiwał w poczekalni, fryzjer nie był zajęty, a w salonie dostępny był wolny fotel.

Kiedy do salonu przychodzi klient, budzi fryzjera (jeżeli jest jakiś śpiący), po czym fryzjer znajduje wolny fotel i obsługuje klienta. Jeżeli nie ma wolnego fotela, klient zajmuje jedno z wolnych miejsc w poczekalni. Jeżeli nie ma miejsca w poczekalni, klient odchodzi.

Należy zsynchronizować prace fryzjerów oraz obsługę klientów w taki sposób, aby jeden fryzjer w danej chwili obsługiwał co najwyżej jednego klienta, jeden klient był obsługiwany przez co najwyżej jednego fryzjera, żaden z klientów ani żaden z fryzjerów nie został zagłodzony.

Zakładamy, że fryzjerzy obsługują kolejnych klientów w kolejności zgłaszania się (i/lub w kolejności zajmowanego miejsca w poczekalni) w miarę dostępnych foteli oraz to że każdy z fryzjerów potrafi wykonać każdą z oferowanych przez salon fryzur.

Przyjmij że czas wykonania każdej z F oferowanych przez salon fryzur wynosi kilka sekund, i że czasy te różnią się dla każdej z fryzur.

Zaimplementuj powyższy symulator, wykorzystując mechanizmy synchronizacji procesów oraz pamięć współdzieloną ze standardu:

1. IPC - System V (60%)
2. IPC - Posix (40%)