|  |  |
| --- | --- |
| Politechnika Białostocka  Wydział Informatyki  Systemy Operacyjne | 06.06.2021 |
| Grupa:  Sokołowski Tomasz PS2  Snarski Jakub PS2  Chomiczewski Gabriel PS2 | Prowadzący:  Mgr. Inż. Tomasz Kuczyński  Ocena:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# Treść zadania

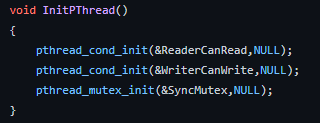
**Czytelnicy i pisarze.** Z czytelni korzysta na okrągło pewna ilość czytelników i pisarzy, przy czym jednocześnie może w niej znajdować się albo dowolna ilość czytelników, albo jeden pisarz, albo nikt - nigdy inaczej. Problem ten ma trzy rozwiązania - z możliwością zagłodzenia pisarzy, z możliwością zagłodzenia czytelników oraz wykluczające zagłodzenie. Napisać:

1. dwa programy symulujące dwa różne rozwiązania tego problemu, bez korzystania ze zmiennych warunkowych [17 p], lub
2. dwa programy symulujące dwa różne rozwiązania tego problemu, przy czym jeden z nich musi korzystać ze zmiennych warunkowych (condition variable). [27 p], lub
3. trzy programy symulujące trzy różne rozwiązania tego problemu, przy czym przynajmniej jeden z nich musi korzystać ze zmiennych warunkowych [34 p].

Ilość wątków pisarzy R i czytelników W można przekazać jako argumenty linii poleceń. Zarówno czytelnicy jak i pisarze wkrótce po opuszczeniu czytelni próbują znów się do niej dostać. Program powinien wypisywać komunikaty według poniższego przykładu:  
  
ReaderQ: 11 WriterQ: 10 [in: R:0 W:1]  
Oznacza to, że w kolejce przed czytelnią czeka 10 pisarzy i 11 czytelników a sama czytelnia zajęta jest przez jednego pisarza. Komunikat należy wypisywać w momencie zmiany którejkolwiek z tych.

# Opis Stworzonych Funkcji

Funkcja void InitPThread() inicjalizuje mutex do synchronizacji oraz wątki zmiennych warunkowych(ReaderCanRead,WriterCanWrite)



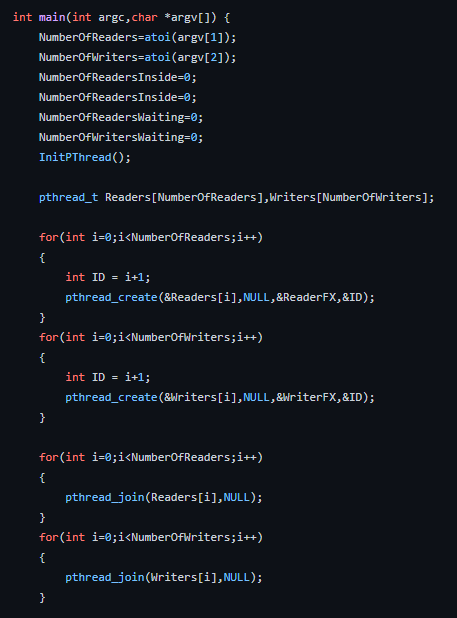
Funkcja void\* ReaderFX(void\* ID) jest funkcją przyjmującą w parametrze wywołanie innego voida z którego może pozyskać numer „czytelnika”, funkcja dla danego wątku czytelnika 100 razy próbuje wejść do czytelni blokując mutex a następnie jeżeli w środku nie ma żadnego pisarza oraz nie ma pisarzy w kolejce wejdzie do czytelni.



Funkcja void\* WriterFX(void\* ID) jest funkcją przyjmującą w parametrze wywołanie innego voida z którego może pozyskać numer „pisarza”, funkcja dla danego wątku czytelnika 100 razy próbuje wejść do czytelni.



Int main(int argc,char \*argvp[]) odpowiada za przypisanie wartości początkowych działania algorytmu w tym za przypisanie oraz stworzenie wątku dla każdego pisarza/czytelnika na podstawie ich ilości podanych w parametrze argv[1] / argv[2] odpowiednio ilość czytelników oraz ilość pisarzy.



# Krótki opis działania programu

Użytkownik inicjalizuje działanie programu po przez wpisanie polecenia ./Conditionals [ilość czytelników] [ilość pisarzy] w konsolę znajdując się w katalogu zawierającym plik wykonawczy.

Program rozpoczyna swoje działanie po przez inicjalizację wątków/mutexu/zmiennych warunkowych

Poszczególne wątki w zależności od zmiennych warunkowych oraz innych wątków „wchodzą”/”wychodzą” z/do czytelni w której każdy przeczekuje wyznaczony czas 1 sekundy. Każda próba wejścia do czytelni jest powielana 100 razy.