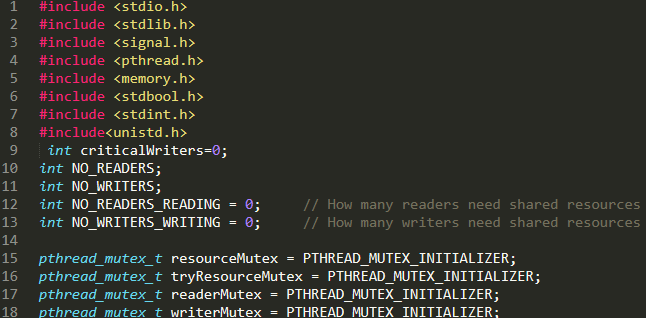
***Projekt: Czytelnicy i pisarze***

*Dominik Makowiak  
Filip Zalewski*

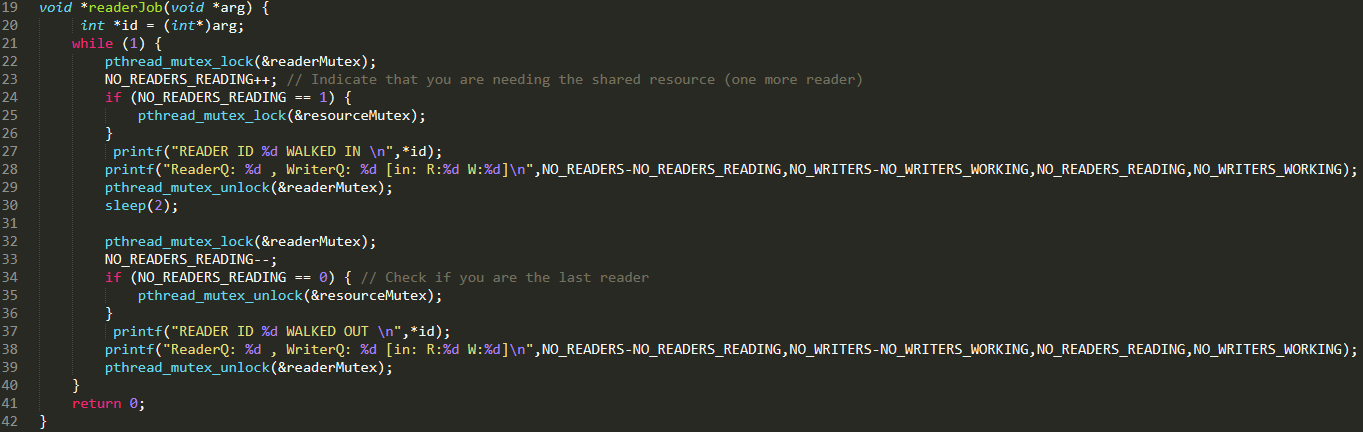
**1. Fragmenty kodu z opisami**



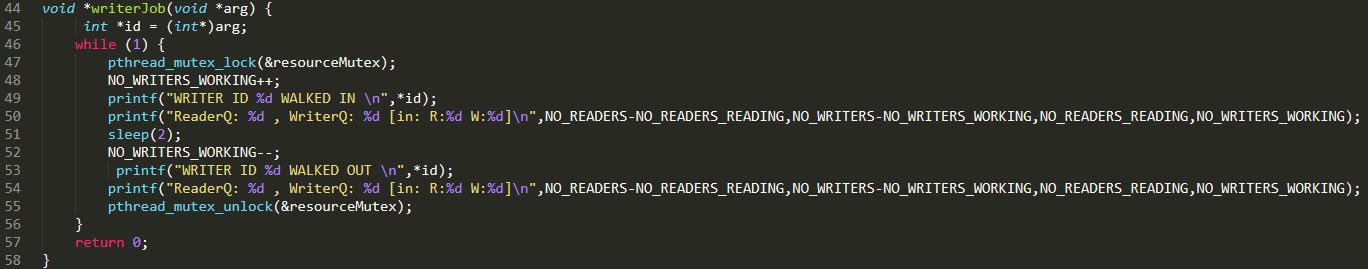
Deklaracja zmiennych globalnych, mutexów oraz bibliotek potrzebnych do wykonania zadania z głodzeniem czytelników. W przypadku rozwiązania z głodzeniem pisarzy nie występują criticalWriters, tryResourceMutex oraz writerMutex, gdyż są zbędne przy tak naiwnym rozwiązaniu.



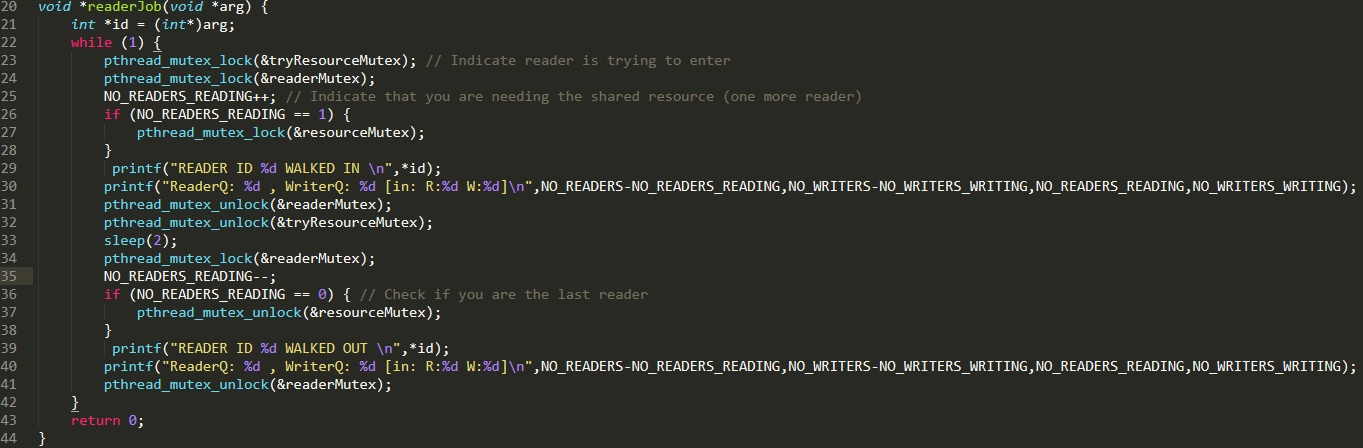
Funkcja main, identyczna dla obu rozwiązań. Można dostrzec typowe dla tego typu zadań zadeklarowanie wątków, utworzenie ich, odczytanie ilości czytelników i pisarzy oraz zakończenie wątków i uwolnienie miejsca zadeklarowanego dynamicznie na korzystanie z wątków.



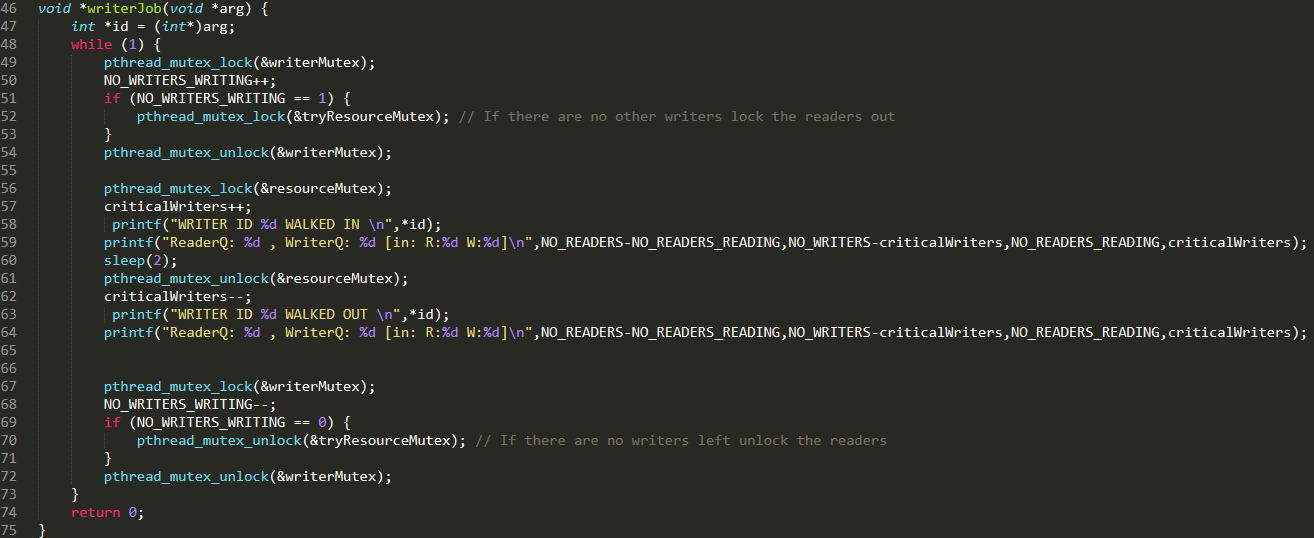
Funkcja odpowiadająca za wpuszczanie czytelników do czytelni w rozwiązaniu zawierającym głodzenie pisarzy. Wyróżnia się ona nakładaniem priorytetu na pisarzy. Można w niej zobaczyć m.in. wprowadzanie wielu pisarzy jednocześnie, jeżeli jeden już znajduje się w środku oraz zwalnianie zasobu w momencie, gdy w środku nie znajduje się żaden pisarz. Pomiędzy dwoma zamknięciami mutexu znajduje się dwusekundowy sleep. W środku można zobaczyć również wykorzystywane w każdej funkcji wypisywanie komunikatów.



Funkcja odpowiadająca za wpuszczanie pisarzy do czytelni w rozwiązaniu zawierającym głodzenie pisarzy. Jak można zauważyć jest ona znacznie mniej rozbudowana niż funkcja odpowiadająca za czytelników w tym rozwiązaniu. Różnica znajduje się również w umiejscowieniu sleepa – jest on wykonywany podczas, gdy mutex jest zamknięty.



Funkcja zbudowana podobnie do tej w rozwiązaniu z głodzeniem pisarzy, lecz różni się koniecznością „zapytania o dostępność zasobu” – w tym rozwiązaniu priorytet mają pisarze, więc czytelnicy muszą pytać o pozwolenie na korzystanie z zasobu.



Funkcja pracy pisarzy rozbudowana bardziej niż w poprzednim rozwiązaniu. Charakteryzuje się nakładaniem priorytetu na pisarzy poprzez blokowanie mutexu „pytania o zezwolenie” na korzystanie z zasobu już przy pierwszym pisarzu w kolejce oraz odblokowująca go dopiero w momencie, gdy nie ma ani jednego w tej samej kolejce. Dodano zmienną criticalWriters, aby dostawać odpowiednie wyniki przy zliczaniu pisarzy w środku czytelni oraz w kolejce.

**2. Sposób użycia**

Aby skompilować program:

gcc -pthread glodzenieczytelnikow.c -o program

gcc -pthread glodzeniepisarzy.c -o program

Aby uruchomić:

*./program ilośćCzytelników ilośćPisarzy*

Aby zakończyć działanie:

*Ctrl + C*