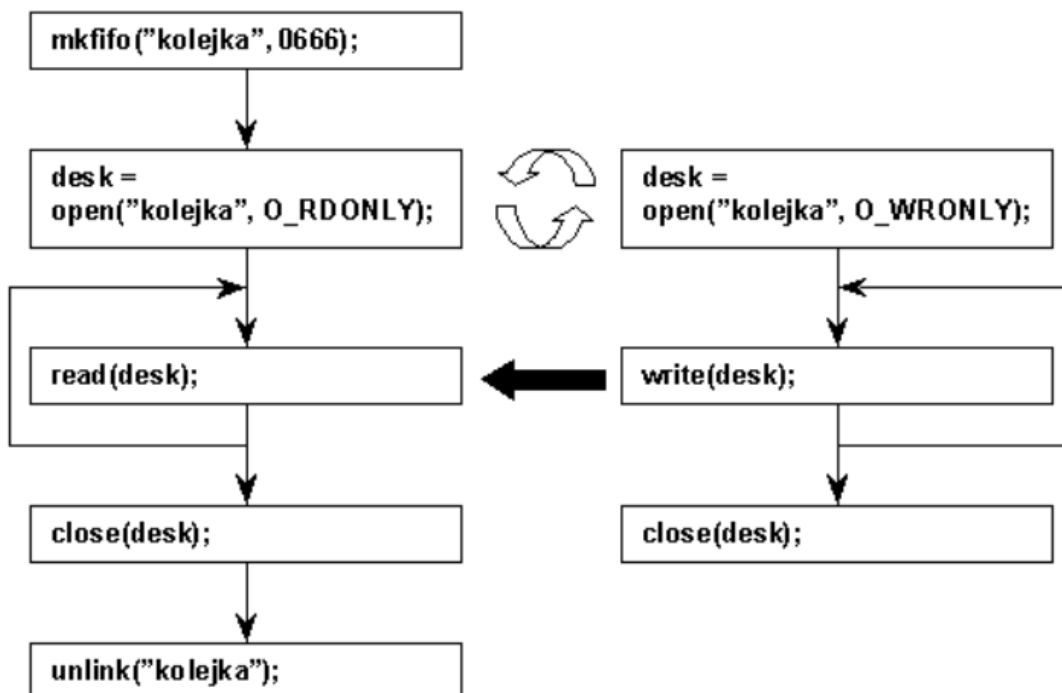

Sprawozdanie 7

Piotr Krześniak (303133)

1. Łącza nazwane w API

Na poprzednim laboratorium poznaliśmy potoki/pipe'y umożliwiające podstawową komunikację międzyprocesową (IPC). Teraz potrzebujemy rozpatrzyć sytuację, aby dwa niezależne procesy komunikowały się ze sobą. Na ratunek przychodzi łącza nazwane, które zwyczajowo są implementowane na kilka sposobów. Opracowane zadania obejmują 3 sposoby. W pierwszym przypadku korzystamy z kolejki **FIFO**.

Mamy do dyspozycji funkcję `mkfifo`, która jako argument przyjmuje po prostu ścieżkę (nazwę pliku) oraz tryb otwarcia pliku - analogia do funkcji `creat`. `mkfifo` tylko tworzy łącza, ale go nie otwiera, do tego celu służy funkcja `open`, przypominając, `open` przyjmuje jako argumenty ścieżkę i trybu otwarcia pliku.



Ponadto wyżej wspomniana funkcja open powoduje zatrzymanie wątku aż nie zostanie otwarty przeciwległy jej koniec. W związku z tym, dodajemy flagę O_NONBLOCK. Jednakże, musimy pamiętać o usunięciu dodanej flagi w funkcji read, bo musimy czekać na treść nadawcy w danym wątku.

System V

Istnieją dwa zestawy system calli używanych do komunikacji, semaforów i blokad. Tym starszym jest właśnie System V. Istnieją trzy zestawy syscalli należące do tej rodziny:

- Message queues - kolejki komunikatów
- Semaphore sets - zestawy semaforów
- Shared memory segments - segmenty pamięci współdzielonej

Nie są to pliki, ani również pliki specjalne (mamy nasz pierwszy wyjątek w unixie). Istnieją one na jednej maszynie, nie da się ich przenieść po sieci, żyją tak długo jak jądro, po wyłączeniu maszyny są niszczone, odwołujemy się do nich po numerycznym identyfikatorze, którego nie da się zmienić przez cały okres życia obiektu, oraz nie posiadają ścieżki i i-node'u

Po analizie dostarczonych plików, wykonaniu testów wszystko w programie działało poprawnie.

POSIX IPC

Standard został wprowadzony w 1993, w pierwszej specyfikacji POSIX, jak część rozszerzeń real-time dla systemu. Mimo najnowszej wersji POSIX (wydanej w 2001, a będącej normą międzynarodową w 2003), jest to nadal rozwiązanie opcjonalne. W porównaniu do System V, syscalli z tej rodziny są wolniejsze, ale wygodnie w użyciu. Obecność System V jest obowiązkowa jeżeli system ma być certyfikowany jako UNIX przez organizację Open Group. Dlatego też, o ile calli POSIX są wygodniejsze, to nie będziemy mieli gwarancji tego, że możemy je zastosować w naszych aplikacjach.

W zadaniu pojawił się błąd, sprawdzając otrzymanego errno wypisuje się błąd EMSGSIZE. Błąd ten spowodowany jest zaniżeniem wartości bufora wiadomości w porównaniu do zawartego w atrybucie mq_msgsize. Sprawdzając domyślny maksymalnego rozmiaru komunikatu dostajemy informacje, że rozmiar wynosi 8192 i zmieniamy w pliku psx.h odpowiadającą mu wartość.