

Szymon Kubiszewski

main.cpp – tworzenie wątków

buff.h – definicja oraz funkcje monitora

prodancons.h – funkcje producentów i konsumentów

monitor.h – plik z serwera studia

buff.h

Bufor dziedziczy po klasie monitora, do synchronizacji wątków używanych jest 8 warunków oraz 2 liczniki ilości parzystych i nieparzystych liczb, posiada też pole będące w istocie samym buforem (std::queue).

Warunki:

do producentów A1:

less_10_even – czy jest mniej niż 10 liczb parzystych

do producentów A2:

more_even – czy jest więcej liczb nieparzystych

do konsumentów B1:

three_nums – czy są co najmniej 3 liczby w buforze

even_head – czy liczba na początku bufora jest parzysta

do konsumentów B2:

seven_nums – czy jest co najmniej 7 liczb w buforze

odd_head – czy liczba na początku bufora jest nieparzysta

Funkcje:

putEven – umieszcza w buforze kolejną liczbę parzystą

putOdd – umieszcza w buforze kolejną liczbę nieparzystą

getEven – usuwa z bufora parzystą liczbę na czele

getOdd – usuwa z bufora nieparzystą liczbę na czele

lookFirst – zwraca liczbę znajdującą się na czele

getEvenCnt – zwraca ilość liczb parzystych w buforze

getOddCnt – zwraca ilość liczb nieparzystych w buforze

show – pokazuje zawartość bufora

put – dodaje liczbę do bufora – wykorzystywane w putOdd/Even

get – usuwa liczbę z bufora – wykorzystywane w getOdd/Even

prodancons.h – funkcje wątków producentów i konsumentów

w pliku tym znajduje się linia #define SEE która została umieszczona w celu ułatwienia wyświetlania działania programu. Kiedy ww. linia jest zakomentowana program będzie wyświetlał zawartość bufora co sekundę np.

0|2|4|1|

2|4|1|

w przeciwnym przypadku:

dodaje 0

dodaje 2

dodaje 4
dodaje 1
wyjmuje 0

Drugi sposób wyświetlania jest niedokładny – mogą wystąpić napisy o wielokrotnym dodaniu danej liczby co jest spowodowane tym, że zmienne wyświetlane jako dodane są wyświetlane już po dodaniu, jednakże samo dodawanie przebiega w sposób prawidłowy, a ten sposób pokazywania działania programu przydawał się szczególnie dla małych ilości producentów.

producentA1 – umieszcza w buforze kolejne liczby parzyste modulo 100, jeśli znajduje się w nim mniej niż 10 liczb parzystych

producentA2 – umieszcza w buforze kolejne liczby nieparzyste modulo 100, jeśli znajduje się w nim więcej liczb parzystych niż nieparzystych

consumerB1 – usuwa z bufora parzystą pierwszą liczbę, jeśli łącznie jest więcej niż 3 liczby

consumerB2 – usuwa z bufora nieparzystą pierwszą liczbę, jeśli łącznie jest więcej niż 7 liczb

main.cpp

Na początku pliku znajdują się 4 zmienne określające ilości producentów i konsumentów.