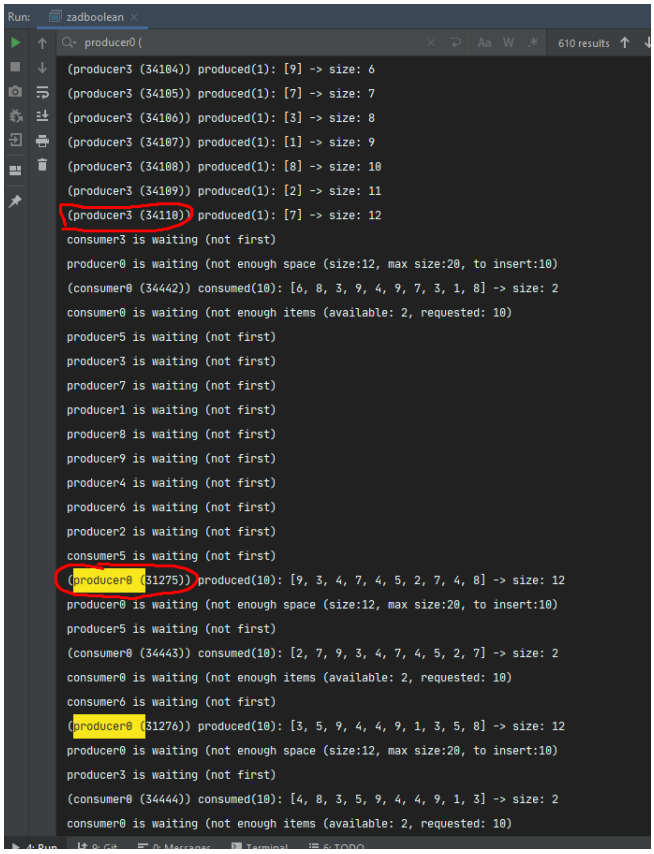
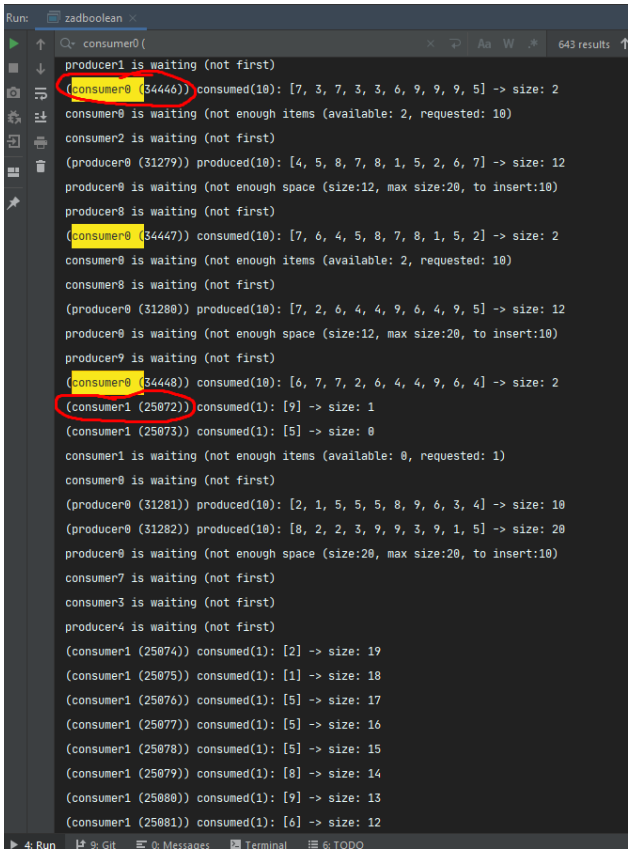
1. Zagładzanie
    a. Zmienna boolean

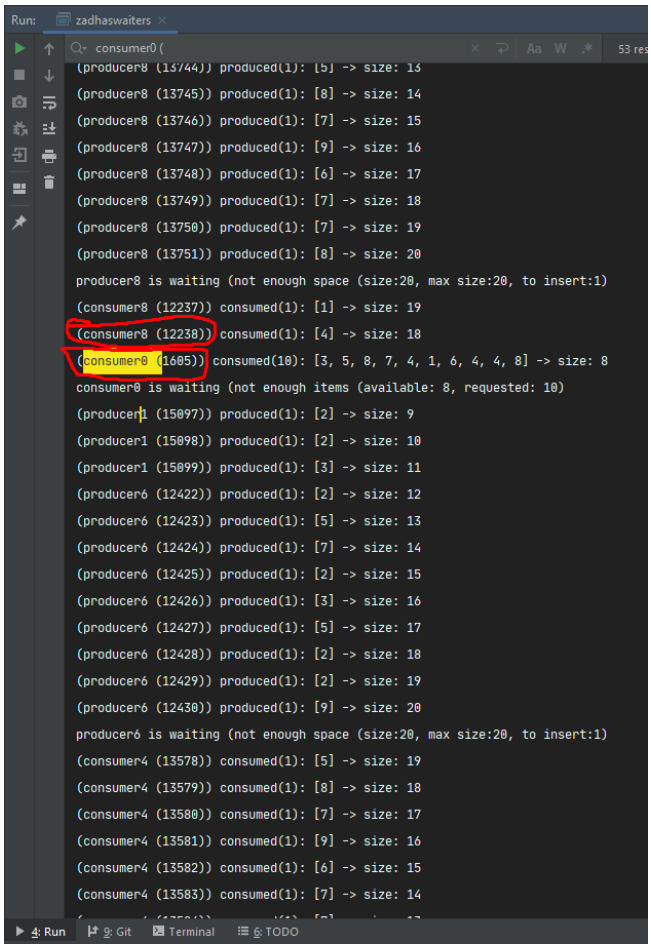
Jeśli używamy zmiennej boolean do zagłodzenia nie dochodzi. Proces A czeka w momencie kiedy nie ma wystarczających zasobów by mógł wykonać swoje zadanie, a reszta procesów (tego samego typu co proces A) czeka aż proces A zakończy wykonywanie zadania i dopiero wtedy następny proces jest dopuszczany do zasobów. Nie ma więc możliwości by proces czekał w nieskończoność na zasoby ponieważ kolejność dostępu do zasobów nie zależy od tego jak wiele ich potrzeba.

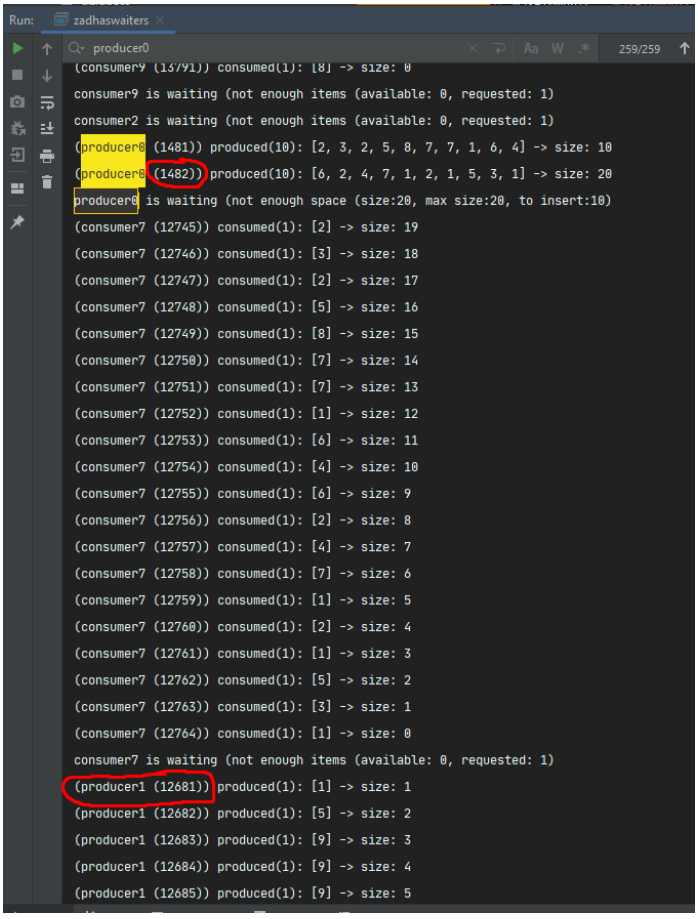




b. Metoda hasWaiters()

Jeśli używamy metody hasWaiters() dochodzi do zagłodzenia (już po chwili różnica w ilości wykonanych operacji to 1 rząd wielkości). Jest to spowodowane tym, że do kolejki pierwszego procesu jest tak naprawdę wpuszczanych więcej procesów, a więc efekt końcowy jest taki sam jak gdybyśmy użyli tylko po jednej kolejce dla każdego typu procesów. Skutkuje to tym, że za każdym razem gdy nie ma dostępnych wystarczających zasobów dla procesu A, trafia do kolejki i inne procesy mogą korzystać z zasobów które były nie wystarczające dla procesu A, co powoduje że przy jego następnym dostępie tych zasobów może znowu brakować.





2. Zakleszczenie
    a. Zmienna boolean

Jeśli używamy zmiennej boolean do zakleszczenia nigdy nie dochodzi, nawet przy dużej różnicy między ilością procesów danych typów.

    b. Metoda hasWaiters()

Używając metody hasWaiters() do zakleszczenia dochodzi za każdym razem. Czas potrzebny na zakleszczenie jest zależny od ilości producentów i konsumentów, np. dla 5 producentów i 1 konsumenta do zakleszczenia dochodzi praktycznie od razu.