

„ZPR PWR – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”

Paradygmaty programowania - ćwiczenia

Lista 12

Wszystkie programy mają być napisane w języku Scala 3.

1. Na wykładzie 12 (str. 10-11) był przedstawiony program producent/konsument z ograniczonym buforem cyklicznym.
 - a) Przepisz ten program, wykorzystując zamiast klasy BoundedBuffer klasę biblioteczną `java.util.concurrent.ArrayBlockingQueue`.
 - b) W programie z podpunktu a) utwórz (w pętli) kilka producentów i konsumentów. Nadaj im unikatowe nazwy, np. `Producer1`, `Consumer1` itd. W jednym z testów utwórz dwa producenty i trzy konsumenty. Dlaczego program się nie kończy? Odpowiedź umieść w komentarzu (koniecznie!).
 - c) Z programu w podpunkcie b) usuń definicje klas `Producer` i `Consumer`. Wykorzystaj `ExecutionContext` do wykonywania odpowiadających im zadań. W jednym z testów utwórz (w pętli) dwa producenty i trzy konsumenty. Dlaczego program się kończy? Odpowiedź umieść w komentarzu (koniecznie!).
2. Zdefiniuj funkcję
`def pairFut[A, B] (fut1: Future[A], fut2: Future[B]): Future[(A, B)] = ???`
 - a) Wykorzystaj metodę `zip` (wykład 12, str. 43)
 - b) Wykorzystaj `for` (wykład 12, str. 44)

Każde zadanie) ma być niezależną aplikacją (o nazwach: Zad1a, Zad1b, Zad1c, Zad2a, Zad2b z testami w metodzie `main`. Wszystkie zadania należy umieścić w pliku `Lista12.scala`. Struktura pliku powinna być podobna do pliku `Traditional.scala` z wykładu.