



## **Laboratorium 2**

### **Wątki**

Studenci w ramach zajęć laboratoryjnych zapoznają się z podstawowymi mechanizmami uruchamiania i synchronizowania wątków.

Zadanie powinno być realizowane jako projekt oparty na narzędziu Apache Maven skonfigurowanym tak aby wykorzystywał platformę Java w wersji 11 lub wyższej. Należy zwrócić uwagę na poprawną identyfikację projektu oraz pakiet wykorzystane w aplikacji.

Należy zaimplementować aplikację w trybie tekstowym pozwalającą na równoległe wykonywanie złożonych obliczeń (np.: naiwna metoda sprawdzania czy duża liczba jest pierwsza). Aby zasymulować złożoność obliczeń można wykorzystać metodę `Thread.sleep` przy mniej czasochłonnych obliczeniach. Rodzaj obliczeń wybiera prowadzący.

Należy zrealizować następujące zadania:

1. Zasób do zgłaszania kolejnych zadań. Należy przygotować współdzielony zasób, który będzie pozwalał na zgłaszanie i pobieranie kolejnych zadań (np.: liczb pierwszych do sprawdzenia). Dodanie nowego zadania nie powinno usuwać poprzedniego jeśli nie zostało jeszcze pobrane. Zasób powinien wykorzystywać mechanizm wait-notify do synchronizacji wątków. (1 pkt)
2. Zasób do zbierania wyników. Należy przygotować współdzielony zasób, do którego będą wprowadzane kolejne wyniki obliczeń. Dodanie kolejnego wyniku nie powinno usuwać poprzednich. Zasób powinien wykorzystywać mechanizmy sekcji krytycznej. (1 pkt)
3. Wątki realizujące obliczenia. Należy przygotować implementację procesu wykonywania zdefiniowanego obliczania, która będzie mogła być przekazana do instancji wątku. Kolejne zadania do wykonania powinny być pobierane z odpowiedniego współdzielonego zasobu w sposób ciągły (wątek po wykonaniu obliczenia wykonuje kolejne). Wyniki obliczeń powinny być zapisywane w odpowiednim współdzielonym zasobie. (1 pkt)
4. Uruchomienie aplikacji. Po uruchomieniu aplikacji należy uruchomić odpowiednią liczbę wątków realizujących złożone obliczenia. Liczba wątków zależy od parametru startowego aplikacji. Użytkownik powinien mieć możliwość zgłaszania kolejnych zadań (np.: poprzez wpisanie polecenia na konsoli odczytanego przez `System.in` za pomocą klasy `Scanner`). (1 pkt)
5. Poprawne zamknięcie aplikacji. Użytkownik powinien mieć możliwość zamknięcia aplikacji (np.: poprzez wpisanie polecenia na konsoli odczytanego przez `System.in` za pomocą klasy `Scanner`). W przypadku zakończenia działania aplikacji należy poprawnie zakończyć działanie wszystkich wątków. (1 pkt)