

Przewodnik 13 – wielowątkowość, tworzenie plików i katalogów

dr inż. Łukasz Sosnowski
Akademia WIT
pod auspicjami Polskiej Akademii Nauk

1 Kontekst

Zadanie polega na stworzeniu programu, który z użyciem puli wątków utworzy 100 plików (pustych) rozlokowanych po maksymalnie 10 katalogach (od 0 do 9). Pliki mają mieć nazwy reprezentujące kolejny numer w danym katalogu. Należy zapewnić synchronizację zasobów wspólnych. Numer katalogu w którym dany wątek ma stworzyć nowy (kolejny) plik podawany jest w konstruktorze w zmiennej `folderNo`. Do zasialania wątków należy zrobić test który będzie tworzył wątki, zarządzał pulą wątków oraz losował numer od 0 do 9 w celu wyboru katalogu dla wątku.

2 W klasie `CounterMultiThread`

a. zaimplementuj interfejs `Runnable` oraz zdefiniuj wymaganą przez niego metodę `run()`.

a. zdefiniuj metodę synchronizowaną zwracającą `String` na podstawie przekazanego numeru folderu. Metoda ma sprawdzać w mapie, czy istnieje już jakiś dodany plik w danym folderze. Jeśli nie to nazwa pliku ma być `1.txt`, jeśli istnieje to należy odczytać aktualną wartość, dokonać inkrementacji i dodać „.txt” i zwrócić taką nazwę.

Wyznaczoną wartość użytą do nazwy pliku należy zapisać w mapie dla danego folderu.

b. w metodzie `run()` należy zaimplementować logikę pobierającą nazwę pliku przy użyciu zdefiniowanej powyżej metody. Należy jednak zrobić to w bloku synchronizowanym (patrz wykład). Następnie po pobraniu nazwy, należy zapewnić implementację tworzącą stosowne katalogi i pliki. Można do tego użyć klasy `File` oraz metod `makedirs()` i `createNewFile()`,

3 W klasie `CounterMultiThreadTest`

Stwórz test, który dla ustalonego katalogu na dysku wykona program z użyciem egzekutora `ExecutorService` z ustaloną pulą wątków na 10 szt. W teście należy wykorzystać klasę `Random` do losowego pobierania nr katalogu dla kolejnych obiektów wątku. W metodzie testowej należy powołać 100 obiektów wątku i z użyciem metody

„execute” wprowadzić je do uruchomienia w puli. Po wykonaniu programu należy zapewnić sprawdzenie poprawności wykonania .np. poprzez policzenie plików w katalogach.