

# Wątki

## Platformy Technologiczne

Michał Piotrowski  
Michał Wójcik

Katedra Architektury Systemów Komputerowych  
Wydział Elektroniki, Telekomunikacji i Informatyki  
Politechnika Gdańska

4 marca 2011

Dwa sposoby tworzenia wątków:

- implementacja interfejsu `Runnable`:
  - pozwala na dziedziczenie po innych klasach;
- dziedziczenie po klasie `Thread`.

Zatrzymanie wątku:

- `Thread.sleep()` — uśpienie wątku na określony czas, brak pewności ile czasu dokładnie wątek będzie uśpiony, możliwe przerwania, pozostaje w sekcji krytycznej, metody statyczna, przerwanie za pomocą `Thread.interrupt()`,
- `Object.wait()` — wychodzi z sekcji krytycznej, metoda instancji, czekanie na spełnienie warunków (`Object.notify()`, `Object.notifyAll()`),
- `Thread.join()` — oczekiwanie na zakończenie innego wątku.

## Obsługa przerwania:

- `catch (InterruptedException e)` — wyjątek rzucony np. przez metodę `sleep`,
- `if (Thread.interrupted())` — wykorzystanie w metodach nie rzucających wyjątku.

## Synchronizacja:

- synchronizacja metod `public synchronized void method() {...}`,
- synchronizacja bloków `synchronized (Object) {...}`.

## Synchronizowane kolekcje:

- `Collection<T>`  
`Collections.synchronizedCollection(Collection<T> c);,`
- `Set<T>` `Collections.synchronizedSet(Set<T> s);,`
- `List<T>` `Collections.synchronizedList(List<T> l);,`
- `Map<K,V>` `Collections.synchronizedMap(Map<K,V> m);,`
- `SortedSet<T>`  
`Collections.synchronizedSortedSet<T>(SortedSet<T> s);,`
- `SortedMap<T>`  
`Collections.synchronizedSortedMap(SortedMap<K,V> m);,`
- w formie dekoratora: `Collections.synchronizedList(new ArrayList<Type>());.`