

# Informatyka w Mechatronice

## Modelowanie typów danych

March 13, 2017

## 1 Cel ćwiczenia

Celem ćwiczenia jest modelowanie i implementacja w języku Java klas reprezentujących dane pomiarowe przesyłane w systemie monitorowania.

## 2 Wymagania

Należy zaprojektować i zaimplementować hierarchię klas odpowiednich do przesyłania danych dotyczących przebiegów czasowych sygnału, widm wąskopasmowych, a także alarmów i alertów.

### 2.1 Wymagane klasy abstrakcyjne

- Cała hierarchia klas zaczyna się od abstrakcyjnej klasy *Packet*, która ma atrybuty o dostępności typu chronionego:
  - device* - nazwa urządzenia będącego źródłem danych
  - description* - opis słowny danych
  - date* - data rozumiana jako ilość sekund od początku czasu, typ *long*
- Abstrakcyjna i szablonowa klasa *Sequence*, z jednym parametrem szablonu, dziedziczy z *Packet* i posiada własne atrybuty o dostępności typu chronionego:
  - channelNr* - numer kanału, typ *int*
  - unit* - jednostka mierzonej wielkości, typ *String*
  - resolution* - rozdzielczość w dziedzinie czasu lub innej, typ *double*
  - buffer* - tablica danych typu T

### 2.2 Wymagane klasy konkretne

Należy zdefiniować następujące klasy konkretne:

- TimeHistory*, która reprezentuje przebieg czasowy. Dziedziczy ona z klasy *Sequence* i posiada własne atrybuty:
  - sensitivity*, reprezentujący czułość napięciową, typ *double*
- Spectrum*, która reprezentuje widmo amplitudowe. Dziedziczy ona z klasy *Sequence* i posiada własne atrybuty:
  - scaling*, oznaczający rodzaj skali (liniowa lub logarytmiczna)
- Alarm*, która reprezentuje widmo amplitudowe. Dziedziczy ona z klasy *Packet* i posiada własne atrybuty:
  - channelNr* - numer kanału, typ *int*
  - threshold* - wartość której przekroczenie powoduje sygnalizację alarmu/alertu
  - direction* - kierunek zmiany (0 - dowolny, -1 - w dół, 1 - w górę)

### 2.3 Wymagane metody w klasach

Klasy powinny mieć konstruktory z parametrami i metodę *toString*, zwracającą łańcuch zawierający wartości atrybutów obiektu.

## 2.4 Testowanie klas

Aby wykonać testowanie klas należy:

1. Utworzyć przykładowe instancje klas *TimeHistory*, *Spectrum* i *Alarm*.
2. Wywołać dla nich metodę *toString*
3. Wyświetlić rezultaty na standardowym wyjściu.