

Zadanie zaliczeniowe 1: *Chmura bytów*

Termin wysyłania rozwiązań upływa 27 listopada 2017 roku o godz. 21:00.

Wprowadzenie

Chmura bytów wymiaru n przyporządkowuje *bytom* miejsca, jednoznacznie identyfikowane przez ciąg n współrzędnych całkowitych. Zachowuje przy tym *niezmiennik chmury*: **każdy byt jest w innym miejscu**.

Chmura bytów może służyć do synchronizacji procesów współbieżnych.

Specyfikacja

W pakiecie `chmura`, implementującym w Javie chmurę bytów wymiaru 2, jest definicja typu `Byt`, kontrolowanego wyjątku `NiebytException` i klasy `Chmura`.

Klasa `Chmura` ma dwa publiczne konstruktory i cztery publiczne obiektowe metody.

- Konstruktor

`Chmura()`

buduje chmurę, która w stanie początkowym nie ma żadnego bytu.

- Konstruktor

`Chmura(BiPredicate<Integer, Integer> stan)`

buduje chmurę, której początkową zawartość określa dwuargumentowy predykat `stan`. W miejscu (x, y) jest byt wtedy i tylko wtedy, gdy `stan.test(x, y)` ma wartość `true`.

- Metoda

`Byt ustaw(int x, int y) throws InterruptedException`

daje jako wynik nowy byt, dodany do chmury w miejscu (x, y) .

- Metoda

`void przestaw(Collection<Byt> byty, int dx, int dy) throws NiebytException, InterruptedException`

przemieszcza na raz wszystkie byty kolekcji `byty` o wektor (dx, dy) . Byt z miejsca (x, y) trafia na miejsce $(x + dx, y + dy)$.

Jeśli którykolwiek z bytów kolekcji `byty` nie jest w chmurze, metoda zgłasza wyjątek `NiebytException`.

Jeżeli wymaga tego niezmiennik chmury, metody `ustaw()` i `przetaw()` wstrzymują wątek do czasu, gdy ich wykonanie będzie możliwe. W przypadku przerwania zgłaszają wyjątek `InterruptedException`.

- Metoda

```
void kasuj(Byte byt) throws NiebytException
```

usuwa `byt` z chmury.

Jeśli `byt` nie jest w chmurze, metoda zgłasza wyjątek `NiebytException`.

- Metoda

```
int[] miejsce(Byte byt)
```

daje dwuelementową tablicę ze współrzędnymi `x` i `y` bytu, lub `null`, jeśli `byt` nie jest w chmurze.

Polecenie

- (8 pkt)

Zrealizuj w Javie chmurę bytów wymiaru 2. Do pakietu `chmura` dołącz wszystkie potrzebne definicje pomocnicze.

- (2 pkt)

Napisz dwa programy przykładowe, demonstrujące zastosowanie chmury bytów do synchronizacji wątków. W pierwszym programie rozwiąż problem producentów i konsumentów a w drugim problem czytelników i pisarzy.

Uwagi

- Program ma być w wersji 8 języka Java. Powinien się kompilować i działać na komputerze `students`.
- Wolno korzystać wyłącznie z pakietów `java.lang` i `java.util`, w tym z `java.util.function` i `java.util.concurrent`.
- Można zignorować fakt, że zakres typu `int` jest ograniczony.
- Implementacja nie musi gwarantować, że wątek nie zostanie zgłodzony.
- Warto zadbać o właściwą obsługę przerwania.
- Rozwiązanie wysyłamy pocztą elektroniczną, jako załącznik do listu, na adres `zaroda@mimuw.edu.pl`. W załączniku powinien być właściwy plik, nie odnośnik do niego, jak to jest w przypadku listów wysłanych przez *USOSmail*.

- Należy wysłać plik o nazwie `ab123456.tar.gz`, gdzie `ab123456` to login na `students`. W paczce `.tar.gz` ma być katalog `chmura` z plikami źródłowymi `.java`.

Walidacja

Rozwiązania zostaną poddane walidacji, wstępnie sprawdzającej zgodność ze specyfikacją.

Na komputerze `students`, w katalogu walidacji, będzie podkatalog `paczki` z rozwiązaniami oraz podkatalog `walidacja` z plikami [waliduj.sh](#) i [WalidacjaChmury.java](#).

Polecenie

```
sh walidacja/waliduj.sh ab123456
```

przeprowadzi walidację rozwiązania studenta o identyfikatorze `ab123456`. Komunikat `OK` informuje o sukcesie.

Rozwiązania, które pomyślnie przejdą walidację, zostaną dopuszczone do testów poprawności.

Na pytania do treści zadania odpowiada [Artur Zaroda](#).