

KURS PROGRAMOWANIA W JAVIE

BUFOR AUTOMATYCZNIE SSĄCY DANE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

Zadanie 1.

Zdefiniuj klasę bufora, który będzie czytywał dane z podanego strumienia tekstowego za pomocą niezależnego wątku. Dane będące obiektami typu `String` mają być gromadzone w kolekcji `ArrayDeque<String>`.

```
public class LineBuffer
{
    private int max;
    private ArrayDeque<String> list = new ArrayDeque<String>();
    private BufferedReader input;

    public LineBuffer (BufferedReader inp, int m) { /*...*/ }

    // ... inne metody
}
```

Praca z tym buforem ma polegać na automatycznym czytaniu i gromadzeniu w kolekcji pośredniczącej `list` danych typu `String`. Gromadzeniem danych ma się zajmować osobny wątek, który będzie przechowywał w jednym momencie nie więcej niż `max` łańcuchów znakowych (jeśli wewnętrzna lista zapełni się do poziomu `max`, to wątek czytający powinien odpocząć). Gdy dane skończą się, lub wystąpi błąd wejścia/wyjścia, to wątek czytający ze strumienia powinien zakończyć swoje działanie.

Wątki zewnętrzne powinny mieć możliwość odczytywania danych z bufora i ich przetwarzania. Gdy bufor jest pusty, ale wątek wewnętrzny czytający strumień jeszcze nie dotarł do końca danych, to należy poczekać na uzupełnienie bufora.

Zadanie 2.

Napisz program konsolowy testujący działanie bufora. Program ten powinien dać się uruchomić na dwa sposoby:

1. bez argumentów — wówczas strumień z danymi ma być skojarzony ze standardowym wejściem;
2. z jednym argumentem — strumień z danymi ma być skojarzony z plikiem, którego nazwę przekazano poprzez argument wywołania;
3. z wieloma argumentami — wówczas strumień z danymi ma być skojarzony ze strumieniem `StringReader` zainicjalizowanym łańcuchem powstałym przez konkatenację argumentów i znaków przejścia do nowej linii (zastosuj obiekt klasy `StringBuffer`).

Główny wątek w programie powinien pracować przetwarzając te dane, wykonując na nich sortowanie przez wstawianie.

Zadanie 3.

Dopisz do bufora komentarze dokumentacyjne. Opisz w nich dokładnie projekt twojej klasy. Na końcu wygeneruj dokumentację dla klasy `LineBuffer` za pomocą programu `javadoc`.

Uwaga.

Dobrze przemyśl projekt zanim zaczniesz programować `LineBuffer`. Pamiętaj o zsynchronizowaniu działania wewnętrznego wątku czytającego dane z zewnętrznymi wątkami odbierającymi te dane (użyj metod `wait()` i `notify()` w blokach synchronizowanych).