laboratorium: zadanie nr 1 *2 października 2007 r.*

KURS PROGRAMOWANIA W JAVIE

LICZBY CAŁKOWITE W POSTACI BINARNEJ

Instytut Informatyki Uniwersytetu Wrocławskiego *Tomasz Wierzbicki i Paweł Rzechonek*

Zadanie 1.

Utwórz plik tekstowy o nazwie `username.java`, gdzie *username* to Twój identyfikator. Plik ten powinien zawierać program w Javie wypisujący do standardowego strumienia wyjściowego `System.out` Twoje imię i nazwisko.

Wskazówka.

Plik `username.java` można utworzyć przy pomocy jakiegokolwiek edytora tekstu. Plik źródłowy kompilujemy poleceniem:

```
unix> javac username.java
```

Kompilator tworzy plik binarny o nazwie `username.class`. Skompilowany program uruchamiamy poleceniem:

```
unix> java username
```

Program w Javie powinien zawierać publiczną klasę o takiej samej nazwie jak nazwa pliku:

```
public class username
{
    treść klasy
}
```

Klasa ta powinna zawierać publiczną statyczną metodę `main`:

```
public static void main (String[] args)
{
    treść metody
}
```

Wykonanie programu polega na wywołaniu tej metody.

Standardowy strumień wyjściowy jest obiektem o nazwie `out` będącym publicznym polem statycznym w klasie `System`. Obiekt `out` posiada między innymi metody `print` oraz `println`, które wypisują do standardowego strumienia wyjściowego napis (obiekt klasy `String`) podany im jako parametr. Metoda `println` powoduje dodatkowo wypisanie znaku nowego wiersza.

Zadanie 2.

Napisz program wypisujący kolejno wszystkie parametry z jakimi zostanie wywołany. Każdy parametr wypisz w osobnym wierszu. Dla przykładu, jeżeli utworzysz program o nazwie `parametry`, to powinien on działać następująco:

```
unix> java parametry abc 123 "Java to język programowania" -3.14
abc
123
Java to język programowania
-3.14
```

Wskazówka.

Parametry programu są umieszczone w tablicy napisów przekazanej metodzie `main` jako parametr. Obiekt tablica posiada pole publiczne `length` zawierające jej rozmiar.

Zadanie 3.

Napisz program wypisujący liczby podane mu jako parametry w postaci dwójkowej. Dla każdego *parametru* program powinien wypisać jeden wiersz postaci

parametr `=` *b*

gdzie *parametr* jest liczbą czałkowitą przekazaną do programu poprzez parametr wywołania, zaś *b* jego przedstawieniem dwójkowym. Jeżeli podany *parametr* nie jest zapisem żadnej liczby, to program powinien wypisać wiersz

`parametr` *parametr* `nie jest poprawnym zapisem dziesietnym liczby`

Program powinien również poprawnie przetwarzać liczby ujemne. Dla przykładu, jeżeli utworzysz program o nazwie `liczby`, to powinien on działać następująco:

```
unix> java liczby 12 34 534534 0 ggg -4324
12 = 1100
34 = 100010
534534 = 10000010100000000110
0 = 0
parametr ggg nie jest poprawnym zapisem dziesietnym liczby
-4324 = -1000011100100
```

Wskazówka.

Do wypisania reprezentacji dwójkowej napisz własną metodę rekurencyjną działającą według następującego schematu:

- w przypadku liczby 0 nic nie wypisuj
- aby wypisać dodatnią liczbę n:
  - wypisz liczbę $n/2$ (tu jest rekursja)
  - wypisz bit $n\%2$

Klasa `Integer` posiada metodę statyczną `parseInt`, która zamienia podany jej napis na liczbę całkowitą. Gdy konwersja jest niemożliwa, to zgłaszany jest wyjątek `NumberFormatException`. Wyjątki obsługuje się przy pomocy instrukcji `try-catch`:

```
try { n = Integer.parseInt(arg[indeks]); }
catch (NumberFormatException ex)
{
    System.out.println ("parametr "+
        arg[indeks]+" nie jest poprawnym zapisem dziesietnym liczby");
}
```