

Lista 10

1. Napisz w Haskellu program, grający z człowiekiem w grę nim (zadanie kontrolne z wykładu 10).
Na początku program powinien zapytać użytkownika o liczbę wierszy (kupek kamieni). Liczba gwiazdek (kamieni) w każdym wierszu ma być wygenerowana losowo (moduł `System.Random`) i równa co najwyżej liczbie wierszy. Po każdym ruchu komputera i człowieka należy wyświetlać planszę. Program nie musi grać optymalnie.

Poniżej są dwa przykłady wykorzystania funkcji z modułu `System.Random`.
Funkcję `randomInts` można wykorzystywać w dowolnych kontekstach, funkcję `randomIntIO` tylko w monadzie `IO`.

Funkcja `randomInts` zwraca listę o długości `n`, zawierającą losowe liczby całkowite z przedziału domkniętego `[lo, hi]`.
Różne zarodki tworzą różne listy.
Funkcja `randomIntIO` wykorzystuje globalny generator liczb losowych z monady `IO`.

```
import System.Random
```

```
randomInts :: Int -> (Int, Int) -> Int -> [Int]
randomInts n (lo, hi) zarodek = take n (randomRs (lo, hi) (mkStdGen zarodek))
```

```
randomIntIO :: (Int, Int) -> IO Int
randomIntIO (lo, hi) = randomRIO (lo, hi)
```

```
example =
  do {raw <- randomIntIO (1, 10);
      howMany <- randomIntIO (1, 10);
      putStrLn (show (raw, howMany))
  }
```