

Programowanie funkcyjne - Kalendarz 2019

Maciej Skwara, Adrian Rymut

08.02.2019

Program Kalendarz 2019 został napisany w języku Haskell. Jest to język czysto funkcyjny.



Programowanie funkcyjne to metodyka programowania będąca odmianą programowania deklaratywnego, w której funkcje należą do wartości podstawowych, a nacisk kładzie się na wartościowanie (często rekurencyjnych) funkcji, a nie na wykonywanie poleceń.

Na samym początku deklarujemy nowe typy zmiennych 'Dzień Tygodnia' oraz 'Miesiąc' wraz z wartościami które im odpowiadają. Używamy do tego funkcji **data**. **deriving** pomaga nam zdefiniować pewne własności nowego typu zmiennej, ponieważ tworząc je, nie mogą być one nawet np. wyświetlone na ekranie. Za to odpowiada klasa typów show.

```
data DzieńTygodnia
  = Poniedziałek | Wtorek | Środa | Czwartek | Piątek | Sobota | Niedziela
  deriving (Eq, Ord, Enum, Bounded, Show)

data Miesiąc
  = Styczeń | Luty | Marzec | Kwiecień | Maj | Czerwiec
  | Lipiec | Sierpień | Wrzesień | Październik | Listopad | Grudzień
  deriving (Eq, Ord, Enum, Bounded, Show)
```

Funkcjami pomocniczymi lecz tak na prawdę funkcje niezbędnymi do odpowiedniego działania programu są **next** i **pad**. Odpowiednio ta pierwsza jest odpowiedzialna za właściwość iteracji po nowych typach zmiennych. Natomiast funkcja **pad** odpowiada za budowę dwuczęściową wartości nowych typów zmiennych.

```
next :: (Eq a, Enum a, Bounded a) => a -> a
next x | x == maxBound = minBound
      | otherwise      = succ x

pad :: Int -> String
pad dzień = case show dzień of
  [c] -> ['0', c]
  sc  -> sc
```

Odpowiada za nią funkcja `miesiąc(m, start, max)`. Wyświetla ona odpowiednio 1. miesiąc - definiowany przy wywoływaniu funkcji 2. '2019' - standardowy ciąg znaków 3. tydzień - standardowy ciąg znaków określony nieco niżej 4. dzień - zmienna określona wartością `Poniedziałek`. Funkcja `dzien` iteruje po kolejnych wartościach zmiennej `DzieńTygodnia` dzięki funkcji `next`. `rok` - wywołuje funkcje `miesiąc` dla różnych wartości `m`, `start` oraz `max`.

```

miesiac :: Miesiac -> DzieńTygodnia -> Int -> String
miesiac m start max = show m ++ " 2019" ++ tydzień ++ dzień Poniedziałek
where
  tydzień = "Su Mo Tu We Th Fr Sa Su"

  dzień aktualny | start == aktualny = dni start 1
                 | otherwise       = " " ++ dzień (next aktualny)

  dni _          m | m > max = "0000"
  dni Niedziela m         = pad m ++ "0" ++ dni Poniedziałek (succ m)
  dni Dzień      m         = pad m ++ " " ++ dni (next dzień) (succ m)

rok = miesiac Styczeń   Worek    31
    ++ miesiac Luty     Piątek   28
    ++ miesiac Marzec   Piątek   31
    ++ miesiac Kwiecień Poniedziałek 30
    ++ miesiac Maj      Środa    31
    ++ miesiac Czerwiec Sobota    30
    ++ miesiac Lipiec   Poniedziałek 31
    ++ miesiac Sierpień Czwartek   31
    ++ miesiac Wrzesień Niedziela   30
    ++ miesiac Październik Worek    31
    ++ miesiac Listopad Piątek     30
    ++ miesiac Grudzień Niedziela   31

```

Program uruchamiamy za pomocą interpretera ghci, używając komendy `:main.`

```

Main> :main
Styczeń 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03 04 05 06
07 08 09 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

Luty 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03
04 05 06 07 08 09 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28

Marzec 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03
04 05 06 07 08 09 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31

Kwiecień 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03 04 05 06 07
08 09 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30

Maj 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03 04 05
06 07 08 09 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31

Czerwiec 2019
Pn Ut Śr Czw Pt Sb Nd
01 02
03 04 05 06 07 08 09
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30

Lipiec 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03 04 05 06 07
08 09 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31

Sierpień 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03 04
05 06 07 08 09 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

Wrzesień 2019
Pn Ut Śr Czw Pt Sb Nd
01
02 03 04 05 06 07 08
09 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30

Październik 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03 04 05 06
07 08 09 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

Listopad 2019
Pn Ut Śr Czw Pt Sb Nd
01 02 03
04 05 06 07 08 09 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30

Grudzień 2019
Pn Ut Śr Czw Pt Sb Nd
01
02 03 04 05 06 07 08
09 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31

```