

Kurs języka Ruby

Lista 3.

Zadanie 1. (3 pkt) Bloki z jednym parametrem można traktować jak definicje jednoargumentowych funkcji. Korzystając z tej obserwacji zaprogramuj dwie procedury. Pierwsza z nich `calka(a,b,&b)` powinna obliczać numerycznie całkę oznaczoną na przedziale $[a, b]$ funkcji zadanej jako blok. Dokładność obliczeń może być ustalona. Druga funkcja to `wykres(a, b, &blok)`, która za pomocą znaków ASCII naszkicuje wykres funkcji danej jako blok. Można przyjąć arbitralny rozmiar terminala.

Implementacje poniższych funkcji powinny być w postaci jednego wyrażenia. Jest to możliwe używając tylko zakresów, operacji na tablicach i bloków. W przypadku bardzo długich wyrażeń akceptowane będzie podzielenie rozwiązania na podwyrażenia.

Zadanie 2. (2 pkt) Napisz jednoargumentową funkcję `pierwsza(n)`, która zwraca tablicę liczb pierwszych nie większych niż n .

Zadanie 3. (2 pkt) Napisz jednoargumentową funkcję `doskonale(n)`, która zwraca tablicę liczb doskonałych nie większych niż n , na przykład

```
doskonale(1000)
==> [6, 28, 496, 8128]
```

Zadanie 4. (2 pkt) Napisz jednoargumentową funkcję `rozklad(n)` która oblicza rozkład liczby n na czynniki pierwsze i zwraca jako wynik tablicę tablic $[[p_1, w_1], [p_2, w_2], \dots, [p_k, w_k]]$ taką, że

$n = p_1^{w_1} * p_2^{w_2} * \dots * p_k^{w_k}$ oraz p_1, \dots, p_k są różnymi liczbami pierwszymi. Na przykład

```
rozklad(756)
==> [[2, 2], [3, 3], [7, 1]]
```

Zadanie 5. (1 pkt) Napisz jednoargumentową funkcję `zaprzyjaznione(n)`, która zwraca tablicę par liczb zaprzyjaźnionych nie większych niż n , na przykład

```
zaprzyjaznione(1300)
==> [[220, 284], [1184, 1210]]
```

Dodatkowe wyjaśnienia można znaleźć w polskiej Wikipedii.

Za rozwiązanie powyższych zadań można uzyskać co najwyżej 4 pkt.

Marcin Młotkowski