

Lista zadań nr 14

Zadania 1–3 rozwiąż w języku Racket, a pozostałe w języku Plait.

Zadanie 1. (2 pkt)

Zdefiniuj strumień S , taki, że $S_n = n!$, wykorzystując procedurę `map2` z wykładu do pomnożenia przez siebie dwóch strumieni. Chyba, że wolisz użyć iteracyjnego podejścia z akumulatorem.

Zadanie 2. (2 pkt)

Zdefiniuj procedurę `partial-sums`, która dla danego strumienia S liczb całkowitych zwróci strumień jego sum częściowych $S_0, S_0 + S_1, S_0 + S_1 + S_2, \dots$. W tym zadaniu z kolei można dodać do siebie dwa strumienie. Albo można użyć akumulatora.

Zadanie 3. (3 pkt)

Rozwiąż problem Hamminga: zbuduj strumień różnych rosnąco uporządkowanych liczb naturalnych, które w rozkładzie na czynniki pierwsze zawierają tylko 2, 3 i 5. Napisz najpierw funkcję `merge`, która scala dwa rosnąco uporządkowane strumienie w jeden, niezawierający powtórzeń, a także funkcję `scale`, która mnoży strumień przez liczbę.

Zadanie 4. (1 pkt)

Wyeliminuj z ewaluatora z pliku `cps.rkt` wszystkie użycia procedury `error`. W tym celu wprowadź typ odpowiedzi obejmujący zarówno wartości jak i informacje o błędzie. Zasygnalizowanie błędu powinno objawiać się zignorowaniem bieżącej kontynuacji i zwróceniem informacji o błędzie. Wystarczy powtórzyć to co zrobiliśmy w maszynie abstrakcyjnej dla języka z błędami na wykładzie 12 (`error-am.rkt`) z tym, że tym razem chcemy by wartości reprezentujące procedury prymitywne również przyjmowały kontynuację i w razie czego sygnalizowały wystąpienie błędu typów – wywołanie `(k (f v2))` powinno zostać zastąpione przez `(f v2 k)`, co oznacza, że typ konstruktora `primopV` musi się odpowiednio zmienić.

Zadanie 5. (2 pkt)

Dodaj do języka z pliku `state-mc.rkt` wyrażenie `{while b e}`, które reprezentuje pętlę *while* znaną z języków imperatywnych: dopóki zachodzi *b*, wykonuj *e*.

Zadanie 6. (2 pkt)

Wyeliminuj użycie funkcji `append` z modelu sterty w pliku `state-store-macros.rkt`. Wprowadź w tym celu jakiś inny sposób adresowania sterty. Można na przykład użyć list par adres i wartość, a `append` zastąpić operacją `cons`, tylko trzeba umieć wygenerować nowy adres (np. jeśli adresy to liczby, to łatwo sprawdzić jaki jest najwyższy używany adres).