

「プログラミング言語」課題

1029-24-9540 山崎啓太郎

May 29, 2013

1 Ex. 3.51

1.1 考え方

`gauche` を使用しているため、`stream.scm` を
http://d.hatena.ne.jp/rucifer_hacker/20090127/1233037207 から使用
した。

1.2 実行例

```
(stream-ref 5)
```

```
(add-load-path ".")  
(load "3.51.scm")
```

```
(stream-ref 5)
```

結果

```
0  
1  
2  
3  
4
```

```
(stream-ref 7)
```

```
(add-load-path ".")
(load "3.51.scm")
```

```
(stream-ref 7)
```

結果

```
0
1
2
3
4
5
6
```

2 Ex. 3.54

2.1 考え方

mul-streams に factorials と (add-streams ones integers) を渡している

2.2 実行例

test/3.54.scm

```
(use gauche.test)
(add-load-path ".")
```

```
(test-start "3.54")
```

```
(load "3.54")
```

```
(test* "(stream-ref factorials 4)" 120 (stream-ref factorials 4))
(test* "(stream-ref factorials 6)" 5040 (stream-ref factorials 6))
```

結果

Testing 3.54 ...

```
test (stream-ref factorials 4), expects 120 ==> ok
test (stream-ref factorials 6), expects 5040 ==> ok
```

3 Ex. 3.58

3.1 考え方

radix は基数だと思われる。
(/ num den) の小数点以下の値が次々と表示される。

3.2 実行例

```
(show-stream (expand 1 7 10) 20)
```

```
(add-load-path ".")  
(load "3.58.scm")
```

```
(show-stream (expand 1 7 10) 20)
```

結果

```
1 4 2 8 5 7 1 4 2 8 5 7 1 4 2 8 5 7 1 4
```

```
(show-stream (expand 3 8 10) 20)
```

```
(add-load-path ".")  
(load "3.58.scm")
```

```
(show-stream (expand 3 8 10) 20)
```

結果

```
3 7 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```