

# 演習問題 5.1(b)

ポテンシャル関数  $\Phi(f, r) = \text{abs}(|f| - |r|)$  を用いて、各操作が  $O(1)$  償却時間となることを示す。

## 操作一覧

```
class Deque q where
  empty :: q a
  isEmpty :: q a -> Bool
  cons :: q a -> a -> q a
  snoc :: q a -> a -> q a
  head :: q a -> a
  last :: q a -> a
  tail :: q a -> q a
  init :: q a -> q a
```

## 各証明

`empty` `isEmpty` `snoc` `head` `last` `init` は省略

### cons

`cons` は実際のステップを1回行う。

ポテンシャル関数は、要素数  $e$  について

- $e = 0$  のとき、 $|f| = 0, |r| = 0$  なので  $\Phi(f, r) = 0$
- $e = 1$  のとき、 $|f| = 1, |r| = 0$  なので  $\Phi(f, r) = 1$
- $e = x (x > 1)$  のとき、 $|f| = 1, |r| = x - 1$  なので  $\Phi(f, r) = x - 2$

となる。このとき、要素追加に伴うポテンシャルの増分は高々  $1$  なので、償却コストは  $2$  である。

償却コストが要素数に関わらず定数で表せるため、`cons` は  $O(1)$  償却時間の処理である。

### tail

連続して `tail` を呼ぶ処理を考えると、`tail` を呼ぶ回数に対して `reverse` を呼ぶ割合が最大となる場合は、`q f r` の `f` が1要素で、`r` が  $|r| = 2^k$  と表せる場合である。

- `f` が 2 要素以上の場合、1要素の場合に比べて割合が小さくなる。
- $|r| = 2^k$  と表せない場合、奇数回 `f` を連続して減らす処理が現れる。これは偶数回の場合に比べて割合が小さくなる。

そこで、`f` が1要素で、`r` が  $|r| = 2^k$  と表せる場合を考える。（要はコストが最も大きくなるパターン）

- `reverse` を呼ばない場合
  - 実際のステップを 1 回行い、`f` を減らすのでポテンシャルの増加は高々  $1$ 、よって償却コストは  $2$
- `reverse` を呼ぶ場合、長さを  $n$  とすると、

- 実際のステップは
  - 要素を減らす 1 回
  - 長さを測定して  $n$  回
  - リストを半分にして  $n/2$  回
  - リストの半分の `reverse` して  $n/2$  回
- ポテンシャルは
  - $n - 1$  から 0 になるので  $n - 1$  減少
- よって償却コストは  $1 + n + n/2 + n/2 - (n - 1) = n + 2$
- 一方、`f` `r` の長さをデータ構造に保持し、 $O(1)$  で参照するようにすると、償却コストは 2 になる

以上から、データ構造の工夫により、`tail` は  $O(1)$  償却時間の処理となる。

## 参考

```
mapM_ (print . foldl cons (empty :: BatchedDeque Int)) [[1..x] | x <- [1..10]]
Q [1] []
Q [2] [1]
Q [3,2] [1]
Q [4,3,2] [1]
Q [5,4,3,2] [1]
Q [6,5,4,3,2] [1]
Q [7,6,5,4,3,2] [1]
Q [8,7,6,5,4,3,2] [1]
Q [9,8,7,6,5,4,3,2] [1]
Q [10,9,8,7,6,5,4,3,2] [1]
```

```
mapM_ print $ iterate tail $ Q [1] [10,9,8,7,6,5,4,3,2]
*** Exception: empty
CallStack (from HasCallStack):
  error, called at Ex1.hs:77:19 in main:PFDS.Sec5.Ex1
Q [1] [10,9,8,7,6,5,4,3,2]
Q [2,3,4,5,6] [10,9,8,7]
Q [3,4,5,6] [10,9,8,7]
Q [4,5,6] [10,9,8,7]
Q [5,6] [10,9,8,7]
Q [6] [10,9,8,7]
Q [7,8] [10,9]
Q [8] [10,9]
Q [9] [10]
Q [] [10]
Q [] []
```