# Solution for Problem Set 7.5

## 201300035 方盛俊

# Problem 1

令 $\hat{c}_i = 2$.

我们要证明 $\sum_{i=1}^{k} c_i \leqslant \sum_{i=1}^{k} \hat{c}_i$ 对于所有的 $k$ 均成立.

使用 Account Method 证明.

Basis: 一开始, 余额值为 $0$.

I.H.: 我们认为, 在第 $i$-th 次操作之前, 余额值 $A$ 大于等于当前栈长度 $L$, 即 $A \geqslant L$.

I.S.:

如果是 `pop()` 操作, 实际开销 $c_i = 1$, 则新余额值 $A' = A + \hat{c}_i - c_i = A + 1$, 新栈长度为 $L' = L - 1$, 则 $A' \geqslant L'$ 仍然成立.

如果是 `insert()` 操作, 我们不妨令 $l$ 为弹出的小于 $x$ 的数据长度, 则实际开销 $c_i = l + 1$, 则余额值 $A' = A + \hat{c}_i - c_i = A - l + 1$, 而新栈长度 $L' = L - l + 1$, 可以看出 $A' \geqslant L'$ 仍然成立.

综上, $A \geqslant L \geqslant 0$

因此均摊开销为 $O(1)$.

# Problem 2

令 $\hat{c}_i = 21$.

我们要证明 $\sum_{i=1}^{k} c_i \leqslant \sum_{i=1}^{k} \hat{c}_i$ 对于所有的 $k$ 均成立.

使用 Account Method 证明.

Basis: 一开始, 余额值为 $0$.

I.H.: 我们认为, 在第 $i$-th 次操作之前, 余额值 $A$ 大于等于当前队列长度 $L$ 的 $20$ 倍, 即 $A \geqslant 20L$.

I.S.:

如果是 `Pull()` 操作, 实际开销 $c_i = 1$, 则新余额值 $A' = A + \hat{c}_i - c_i = A + 20$, 新队列长度为 $L' = L - 1$, 则 $A' \geqslant 20L'$ 仍然成立.

如果是 `Push()` 操作, 实际开销 $c_i = 1$, 则新余额值 $A' = A + \hat{c}_i - c_i = A + 20$, 新队列长度为 $L' = L + 1$, 则 $A' \geqslant 20L'$ 仍然成立.

如果是 `Size()` 操作, 实际开销 $c_i = 1$, 则新余额值 $A' = A + \hat{c}_i - c_i = A + 20$, 新队列长度为 $L' = L$, 则 $A' \geqslant 20L'$ 仍然成立.

如果是 `Decimate()` 操作, 因为要进行 $L$ 次循环, 且每一次最多会执行一次 `Push()` 和一次 `Pull()`, 实际开销 $c_i \leqslant 2L \leqslant 0.1A + 21$, 我们取 $c_i = 0.1A + 21$, 则余额值 $A' = A + \hat{c}_i - c_i = A + 21 - 0.1A - 21 = 0.9A$, 而新栈长度 $L' = L - 0.1L = 0.9L$, 可以看出 $A' \geqslant L'$ 仍然成立.

综上, $A \geqslant 2L \geqslant 0$

因此均摊开销为 $O(1)$.