# Дек

Санду Р.А.

26 ноября 2017 г.

Будем реализовывать дек на зацикленном динамическом массиве. Для этого, помимо нужно поддерживать 3 значения: размер динамического массива $n$, размер дека $s$, индекс начала дека $b$. Сам дек обозначим $D$, а элементы динамического массива $A_i$. Также будем поддерживать инварианты: $n$ является степенью двойки, $1 \le n$, $\frac{n}{4} < s < n$. При нарушении инвариантов, будем либо увеличивть $n$ в 2 раза, либо в 2 раза уменьшать. Для простоты реализации, меньше единицы $n$ не будем делать никогда.

Операция Get:

Get($i$)

// Значение возвращается по ссылке, т.е. с возможностью изменения
**return** $A_{(b+i)\%n}$

Операция Restore-Invariants:

Restore-Invariants()

**if** $s == n$
    // Выделение нового массива $B$ размера $2n$
    // Перемещение элементов Get($0 \mathinner{..} s-1$) в $B_{0..s-1}$
    $A = B$
    $b = 0$
**elseif** $4s \le n$ and $1 \le \frac{n}{2}$
    // Выделение нового массива $B$ размера $\frac{n}{2}$
    // Перемещение элементов Get($0 \mathinner{..} s-1$) в $B_{0..s-1}$
    $A = B$
    $b = 0$
    $n = \frac{n}{2}$

Операция Push-Back:

Push-Back($v$)

Get($size$) $= v$
$s = s + 1$
Restore-Invariants()

Операция Push-Front:

Push-Back($v$)

```
// Подразумевается "правильное"математическое определение взятия по модулю
b = (b − 1)%n
Get(0) = v
s = s + 1
Restore-Invariants()
```

Операция Pop-Back:

Pop-Back()

```
s = s − 1
Restore-Invariants()
```

Операция Pop-Front:

Pop-Back()

```
b = b + 1
s = s − 1
Restore-Invariants()
```

Проведём амортизационный анализ этих операций методом учётных стоимостей.

Не сложно заметить, что операции Push-Back и Push-Front, Pop-Back и Pop-Front имеют одинаковый с точки зрения асимптотики принцип работы, поэтому будем рассматривать операции Push и Pop, которые работают за $O(1)$, если не нарушили инвариант, и за $O(s)$ иначе. Положим реальные стоимости обоих операций равными 1. Также довольно целесообразно рассматривать индексацию этого индексированного массива так, как она указана в операции Get.

Операция Push, не учитывая нарушения инвариантов. Положим её амортизированную стоимость равной 3. При добавлении элемента будем класть 1 из монет на сам добавленный элемент, одну монету будем тратить на выполнение самой операции, а оставшуюся монету будем класть на любой из элементов, на котором лежит минимальной кол-во монет.

Операция Pop, не учитывая нарушения инвариантов. Её амортизированная стоимость пусть будет равна 2. Тогда при удалении элемента будем одну монету тратить на выполнение операции, а одну из них класть аналогично последней монете для Push.

Оплаченность операции по востановлению инварианта будем доказывать по индукции. База — при создании стека всё оплачено. Переход. Пусть инвариант только что был востановлен. Тогда $s = 2^k$. Из вышенаписанного следует, что до момента нарушения инвариантов все операции оплачены.

Пусть мы совершили операцию Pop и нарушили инвариант. Тогда с момента востановления инварианта, после которого у нас могло быть суммарно 0 монет на всех элементах, прошло не меньше $2^{k-1}$ операций, а следовательно на каждом из $2^{k-1}$ оставшихся элементов лежит хотя бы по одной монете, которые мы и используем на перемещение этих элементов в новый массив.

Пусть мы совершили операцию Push и нарушили инвариант. Тогда с момента востановления инварианта, прошло как минимум $2^k$ операций Push, а значит на каждом из $2^{k+1}$ элементов лежит как минимум одна монета, а значит перемещение всех элементов оплачено.

Таким образом, шаг индукции завершён: после востановления инварианта суммарное кол-во монет не станет отрицательным и все операции оплачены. Получаем амортизированную сложность всех операций $O(1)$.

Проведём амортизационный анализ методом потенциалов.

Будем рассматривать последовательность операций и состояний $D_i$, где $c_i$ — стоимость операции из состояния $D_i$ в $D_{i+1}$, а $\hat{c}_i$ — её амортизированная стоимость. Возьмём следующую функцию потенциала

$$\Phi(D) = \begin{cases} 2s - n & \left\lfloor \frac{1}{2}n \right\rfloor < s \\ \left\lfloor \frac{1}{2}n \right\rfloor - s & s \leq \left\lfloor \frac{1}{2}n \right\rfloor \end{cases}$$

Для неё по определению

$$\forall i. \Phi(D_0) = 0 < \Phi(D_i)$$

а следовательно функция потенциала определена корректно.

Найдём амортизированную стоимость всех операций

$$\hat{c}_i = c_i + \Phi(D_{i+1}) - \Phi(D_i).$$

Округление вниз в формулах опустим, т.к. оно играет роль только при $n = 1$, а в этом тривиальном случае все операции точно занимают константное время. Для Push:

1й случай — $s < \frac{1}{2}n$

$$\hat{c}_i = c_i + (\frac{1}{2}n - s - 1) - (\frac{1}{2}n - s) = 0$$

2й случай — $s = \frac{1}{2}n$

$$\begin{aligned} \hat{c}_i &= c_i + (2s + 2 - n) - (\frac{1}{2}n - s) \\ &= c_i + 2 + 3s - \frac{3}{2}n \\ &= c_i + 2 + 3\frac{n}{2} - \frac{3}{2}n \\ &= c_i + 2 = 3 \end{aligned} \tag{1}$$

3й случай — $\frac{1}{2}n < s$

$$\hat{c}_i = c_i + (2s + 2 - n) - (2s - n) = 3$$

4й случай — $s + 1 = n$

$$\begin{aligned}\hat{c}_i &= c_i + (\frac{1}{2}2n - (s+1)) - (2s - n) \\ &= c_i + (n - n) - (2(n-1) - n) \\ &= c_i - n + 2 = \\ &= n - n + 2 = 2\end{aligned} \tag{2}$$

Получаем константную стоимость во всех случаях, а следовательно амортизированную асимптотику $O(1)$. Для Pop:

1й случай — $s + 1 = \frac{1}{4}n$

$$\begin{aligned}\hat{c}_i &= c_i + (\frac{1}{4}n - (s+1)) - (\frac{1}{2}n - s) \\ &= c_i + (\frac{1}{4}n - \frac{1}{4}n) - (\frac{1}{2}n - \frac{1}{4}n - 1) \\ &= c_i - \frac{1}{4}n + 1 = \\ &= \frac{1}{4}n - \frac{1}{4}n + 1 = 1\end{aligned} \tag{3}$$

2й случай — $s \leq \frac{1}{2}n$

$$\hat{c}_i = c_i + (\frac{1}{2}n - s + 1) - (\frac{1}{2}n - s) = 2$$

3й случай — $s - 1 = \frac{1}{2}n$

$$\begin{aligned}\hat{c}_i &= c_i + (\frac{1}{2}n - s + 1) - (2s - n) \\ &= c_i - 1 - 3s + \frac{3}{2}n \\ &= c_i - 1 + 3\frac{n}{2} + 3 - \frac{3}{2}n \\ &= c_i + 2 = 3\end{aligned} \tag{4}$$

4й случай — $\frac{1}{2}n < s - 1$

$$\hat{c}_i = c_i + (2s - 2 - n) - (2s - n) = -1$$

Таким образом, получаем константную стоимость во всех случаях, а значит амортизированную асимптотику $O(1)$.