تحلیل سرشکنی (Amortized Analysis)

- تحلیل بدترین حالت
- تحلیل در حالت متوسط

محیط: یک داده‌ساختار و دنباله‌ای از عملیات بر روی آن

هزینه‌ی سرشکن‌شده هر عمل: متوسط هزینه‌ی آن عمل (در بدترین حالت)

مثال ۱: پشته با عمل multiPoP

علاوه بر Push و Pop

```
MULTIPOP(S, k)
1  while not STACK-EMPTY(S) and k ≠ 0
2       do POP(S)
3          k ← k − 1
```

هزینه‌ها در بدترین حالت:

- $\textsc{Push}$ و $\textsc{Pop}$: $\Theta(1)$
- $\textsc{Multipop}(S,k)$: $\Theta(\min\{k, \textsc{Size}(S)\})$

پس اگر دنباله‌ای از n تا از این اعمال داشته باشیم،
جمع کل هزینه‌ها در بدترین حالت: $\Theta(n^2)$.

نشان می‌دهیم که هزینه‌ی سرشکن‌شده هر عمل $O(1)$ است.

مثال ۲: افزایش شمارنده‌ی دودویی

یک شمارنده‌ی دودویی k بیتی $A[۰..k-۱]$

(بیت ۰ کم‌ارزش‌ترین بیت)

```
INCREMENT(A)
i ← 0
while i < length[A] and A[i] = 1
      do A[i] ← 0
         i ← i + 1
if i < length[A]
  then A[i] ← 1
```


© محمد قدسی

هزینه‌ی شمارنده در بدترین حالت

هر عمل افزایش در بدترین حالت $\mathcal{O}(k)$،

n عمل افزایش $\mathcal{O}(nk)$

نشان می‌دهیم که هزینه‌ی هر عمل افزایش به‌صورت سرشکن‌شده $\mathcal{O}(۱)$ است.

روش‌های تحلیل سرشکن‌شده

- روش انبوهه (aggregate): جمع هزینه‌های اعمال، تقسیم بر تعداد.
- روش حساب‌داری (accounting): استفاده از یک مخزن پول و پرداخت برای هر عمل
- روش تابع پتانسیل (potential): حالت کلی‌تر روش قبل

تحلیل پشته با روش مجموع

تحلیل پشته با روش مجموع

اگر n عمل داشته باشیم:

- حداکثر تعداد عناصر پشته n است.
- هر عنصر داخل پشته دقیقاً یک بار Push و حداکثر یک بار Pop می‌شود.
- یک عنصر یا مستقیماً Pop می‌شود و یا با Multipop
- هر عمل Push و Pop $\mathcal{O}(۱)$ است.

جمع هزینه‌ها حداکثر $\Theta(۲n)$ است

پس هزینه‌ی سرشکن‌شده‌ی هر عمل $\mathcal{O}(۱)$ است.

تحلیل شمارنده با روش انبوهه

تحلیل شمارنده با روش انبوهه

- هر عمل افزایش حداکثر یک بیت را ۱ می‌کند و تعدادی را از ۱ به ۰ تغییر می‌دهد
- بیت ۰ با هر افزایش flip می‌شود (n بار)
- بیت ۱ یک در میان flip می‌شود ($n/۲$ بار)
- بیت ۲ هر ۴ بار یک دفعه تغییر می‌کند ($n/۴$)
- ...
- بیت i پس از ۲^i بار افزایش تغییر می‌کند (در مجموع $\lfloor \frac{n}{۲^i} \rfloor$ بار)

پس در مجموع

$$\sum_{i=0}^{k-1} \lfloor \frac{n}{2^i} \rfloor \leq n \sum_{i=0}^{\infty} \frac{1}{2^i} \leq 2n$$

یعنی هزینه‌ی سرشکن‌شده‌ی هر عمل $\mathcal{O}(1)$ است.

تحلیل پشته به‌روش حساب‌داری

- برای هر Push ۲ ریال خرج می‌کنیم.
- ۱ ریال صرف عمل Push می‌شود.
- ۱ ریال را بر روی عنصر داخل پشته می‌گذاریم.
- Pop ها همه مجانی خواهند بود.

کلاً $2n$ ریال هزینه پس هزینه‌ی سرشکن‌شده $\mathcal{O}(1)$ است.

تحلیل شمارنده به‌روش حساب‌داری

تحلیل شمارنده به‌روش حساب‌داری

- هر عمل افزایش ۲ ریال هزینه می‌کنیم.
- ۱ ریال صرف تبدیل یک بیت از ۰ به ۱ می‌شود.
- ۱ ریال برروی بیت ۱ قرار می‌دهیم.
- هزینه‌ی ۱ به ۰ مجانی خواهد بود.

روش تابع پتانسیل

- D_0 : داده‌ساختار اولیه
- D_i: داده‌ساختار پس از عمل i ام

$$D_0 \xrightarrow{\text{عمل } 1} D_1 \xrightarrow{\text{عمل } 2} D_2 \ldots \xrightarrow{\text{عمل } n} D_n$$

- c_i: هزینه‌ی واقعی عمل i ام
- تعریف می‌کنیم: $\Phi(D_i) =$ تابع پتانسیل که D_i به عدد حقیقی نگاشت می‌کند.

- **تعریف: $\hat{c}$ هزینه‌ی سرشکن‌شده‌ی عمل i ام**

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

- **پس** $\sum_{i=1}^{n} \hat{c}_i = \sum_{i=1}^{n} c_i + \Phi(D_n) - \Phi(D_0)$
- **اگر** $\Phi(D_0) = 0$،

$$\sum_{i=1}^{n} \hat{c}_i \geq \sum_{i=1}^{n} c_i$$

یعنی $\sum_{i=1}^{n} \hat{c}_i$ کران بالایی برای $\sum_{i=1}^{n} c_i$ است که می‌خواهیم به‌دست آوریم.

تناظر با روش حساب‌داری

- $\Phi(D_i)$: مقدار پول موجود در مخزن پس از عمل i.
- $\hat{c}_i$: مقدار پولی که برای عمل i پرداخت می‌کنیم.
- c_i: مقدار هزینه‌ی صرف‌شده برای عمل i.
- اگر $c_i < \hat{c}_i$، مابه‌التفاوت به مخزن اضافه می‌شود: $\Phi(D_i) = \Phi(D_{i-1}) + \hat{c}_i - c_i$
- اگر $c_i > \hat{c}_i$، برای انجام عمل i لازم است از مخزن به‌اندازه‌ی $c_i - \hat{c}_i$ برداریم تا بتوانیم این عمل را انجام دهیم. پس پول مخزن $\Phi(D_i) = \Phi(D_{i-1}) - (c_i - \hat{c}_i)$ می‌شود.

تحلیل پشته با روش پتانسیل

- تعریف: تعداد عناصر موجود در پشته $= \Phi(D_i)$
- در ابتدا $\Phi(D_0) = 0$
- عمل Push:
  - یک عنصر اضافه می‌شود: $\Phi(D_i) - \Phi(D_{i-1}) = 1$
  - هزینه‌ی واقعی $c_i = 1$،
  - پس $\hat{c}_i = 1 + 1 = 2$.

- عمل Pop:
  - یک عنصر کم می‌شود: $\Phi(D_i) - \Phi(D_{i-۱}) = -۱$
  - هزینه‌ی واقعی $c_i = ۱$،
  - پس $\hat{c}_i = ۱ - ۱ = ۰$.
- عمل Multipop هم تعدادی Pop است. پس هزینه‌ی سرشکن‌شده‌ی آن هم صفر است.

$$\sum_{i=۱}^{n} c_i \leq \sum_{i=۱}^{n} \hat{c}_i = ۲n$$

پس هزینه‌ی سرشکن‌شده‌ی هر عمل $\mathcal{O}(۱)$ است.

تحلیل شمارنده به‌روش پتانسیل

- تعریف تابع پتانسیل: تعداد یک ها A $\Phi(D_i) =$
- داریم $\Phi(D_\circ) = \circ$ و $\Phi(D_i) \geq \circ$
- عمل «افزایش»:
  - t_i بیت از ۱ به صفر و حداکثر یک عدد صفر به ۱ تبدیل می‌شود.
  - پس

$$\left.\begin{array}{l} \hat{c}_i = \Phi(D_i) - \Phi(D_{i-۱}) + c_i \\ \Phi(D_i) - \Phi(D_{i-۱}) \leq -t_i + ۱ \\ c_i \leq t_i + ۱ \end{array}\right\} \rightarrow \hat{c}_i \leq ۲$$

• پس هزینه‌ی سرشکن‌شده‌ی هر عمل افزایش $\mathcal{O}(۱)$ است.