دکتر آبام
علی مقدسی

دانشکده مهندسی کامپیوتر
طراحی الگوریتم‌ها

نیمسال دوم ۱۴۰۵-۱۴۰۴
۴۰۲۱۰۶۵۴۲

تمرین سوم

# سوال ۱۶ – بیت‌فلیپ

صورت سوال

یک صفحه‌ی $n \times m$ از لامپ‌ها داریم. هر لامپ یا خاموش است یا روشن.

در هر عملیات می‌توانیم دقیقاً یکی از کارهای زیر را انجام دهیم:

- وضعیت همه‌ی لامپ‌های یک سطر را عوض کنیم؛ یعنی لامپ‌های روشن خاموش شوند و لامپ‌های خاموش روشن شوند.
- وضعیت همه‌ی لامپ‌های یک ستون را عوض کنیم.

می‌خواهیم با کمترین تعداد عملیات، همه‌ی لامپ‌ها را روشن کنیم.

یک الگوریتم با پیچیدگی زمانی $O(nm)$ برای محاسبه‌ی کمترین تعداد عملیات لازم ارائه دهید.

## مدل‌سازی

روشن = 1، خاموش = 0. فرض می‌کنیم $r_i \in \{0,1\}$ یعنی «سطر $i$ flip شود؟» و $c_j \in \{0,1\}$ یعنی «ستون $j$ flip شود؟».
هر خانه $(i,j)$ دقیقا $r_i + c_j$ بار $(2 \bmod)$ flip می‌شود. پس:

$$a_{ij} \oplus r_i \oplus c_j = 1 \quad \iff \quad r_i \oplus c_j = \texttt{need}_{ij}$$

که `need[i][j]` = 1 اگر چراغ خاموش است و 0 اگر روشن است.
**هدف:** کمینه کردن $\sum r_i + \sum c_j$.

## الگوریتم ($O(nm)$)

۱. برای همه خانه‌ها، `need[i][j]` را می‌سازیم ($O(nm)$).

۲. **شرط سازگاری:** برای همه $i, j$:

$$\texttt{need[i][j]} \oplus \texttt{need[0][j]} = \texttt{need[i][0]} \oplus \texttt{need[0][0]}$$

اگر نقض شد ⇒ **جوابی وجود ندارد** (با هیچ ترکیبی همه روشن نمی‌شوند).

۳. یک جواب را با $r_0 = 0$ بساز:

- $c_j = \texttt{need[0][j]}$
- $r_i = \texttt{need[i][0]} \oplus c_0$

(برای $i \geq 1$)

اگر سازگاری برقرار باشد، این جواب درست است.

۴. $S = \sum r_i + \sum c_j$ را حساب کن.

۵. **نکته:** اگر همه $r_i$ و $c_j$ را برعکس کنیم ($0 \to 1$)، هنوز معادلات برقرارند. پس دو جواب داریم:

$$\text{جواب} = \min\left(S,\ n + m - S\right)$$

**مثال کوچک** ($2 \times 2$، همه خاموش):

$$\texttt{need} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

با $r_0 = 0$: $c = [1,1]$، $S = 2 \Rightarrow r = [0,0]$. چون $n + m - S = 2$ **هم هست**، جواب $= 2$.

## درستی

اگر سطر $0$ را flip کنیم یا نکنیم، ستون‌ها یکتا تعیین می‌شوند. بعد هر $r_i$ هم از ستون $0$ مشخص می‌شود. شرط سازگاری همان است که این انتخاب با همه خانه‌ها جور دربیاید.
دو جواب مکمل (همه را برعکس کنیم) هزینه $S$ و $n + m - S$ می‌دهند؛ پس کمینه کافی است.

## پیچیدگی

- ساخت need + بررسی سازگاری: $O(nm)$
- ساخت جواب و شمارش: $O(n + m)$
- **مجموع:** $O(nm)$ ✓