



DINAMIKUS PROGRAMOZÁS

2016. november 4-6.

Kiss Elemér Kreatív matematika és
programozói tábor

Kőszáli kecske

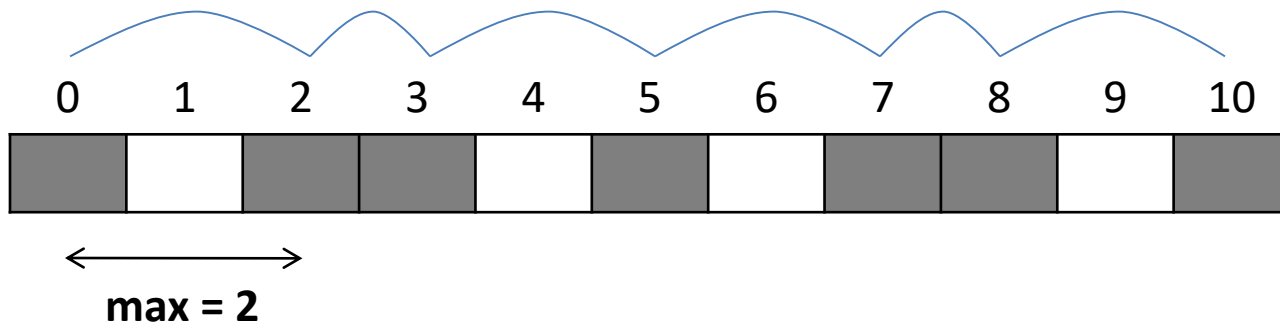
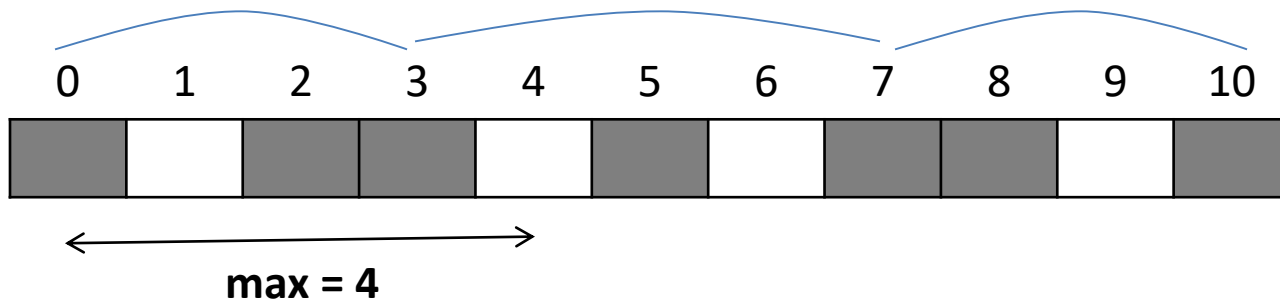
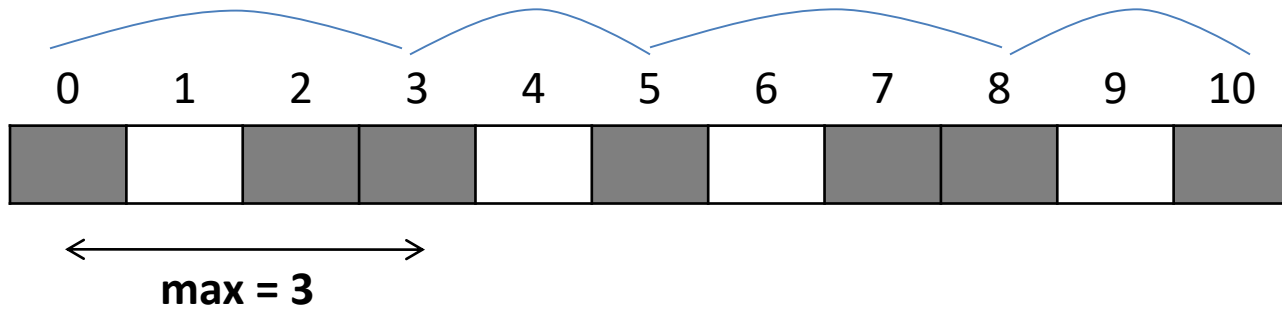
Székelykeresztúr



Kőszáli-kecske – 1

- $(n+1)$ hosszú híd
 - Adott, hogy milyen pozíciókban vannak deszkák
 - A 0. és n . pozíciókban garantáltan van deszka
- Adott a maximális távolság, amit a kőszáli kecske meg tud ugorni
- KÉRDÉS:
 - Mennyi a minimális számú ugrás, amivel a kőszáli kecske át tud jutni a hídon?
 - Feltéve, hogy létezik átjutási lehetőség

Kőszáli-kecske – 1

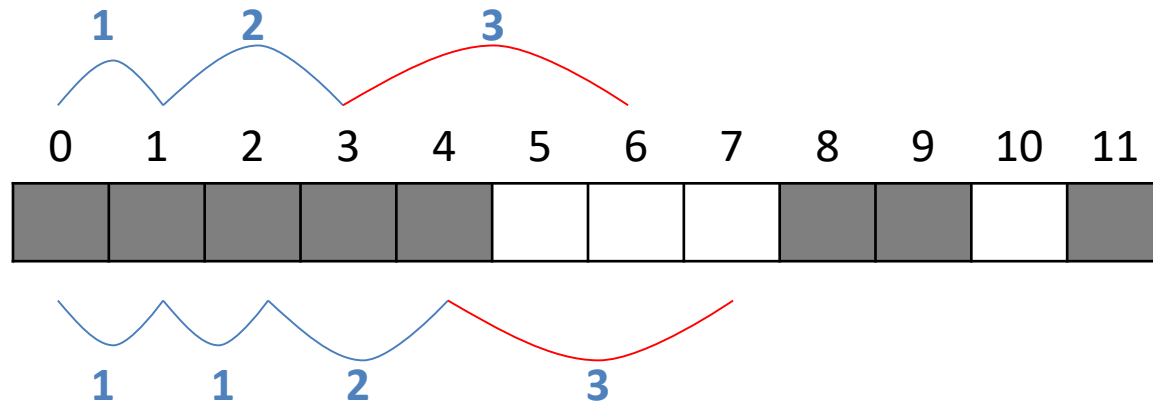
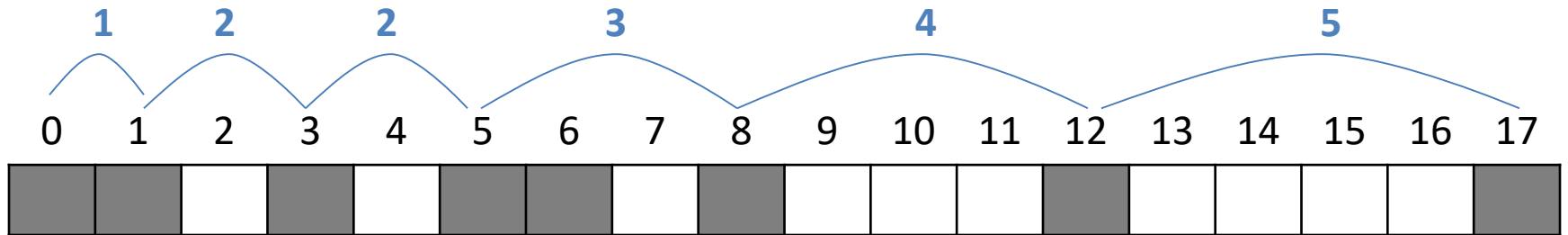


Kőszáli-kecske – 2

- Adott, hogy milyen pozíciókban vannak deszkák
 - A 0. és n. pozíciókban garantáltan van deszka
 - Az 1. ugrás kötelezően 1 hosszú kell, hogy legyen
 - Ha a kurrens ugrás k hosszú volt, akkor a következő kötelezően (k-1), k vagy (k+1) hosszú, kell hogy legyen
- KÉRDÉS:
 - **Át tud-e jutni a kőszáli kecske a hídon?**

$$1 < \text{hídhossz} < 2^{31}$$
$$1 < \text{deszkaszám} < 1100$$

Köszáli-kecske – 2



Kőszáli-kecske – 2

Mikor tudja megugrani a kecske az i . és j .
pozíciójú deszkák közti, nyilván $(j-i)$ hosszú
távolságot?

Kőszáli-kecske – 2

Mikor tudja megugrani a kecske az i . és j .
pozíciójú deszkák közti, nyilván $(j-i)$ hosszú
távolságot?

Ha az i . deszkára $(j-i)-1$, $(j-i)$ vagy $(j-i)+1$ hosszú
ugrásokkal tudott érkezni

Kőszáli-kecske – 2

Mikor tudja megugrani a kecske az i . és j . pozíciójú deszkák közti, nyilván $(j-i)$ hosszú távolságot?

Ha az i . deszkára $(j-i)-1$, $(j-i)$ vagy $(j-i)+1$ hosszú ugrásokkal tudott érkezni

Vagyis ha előzőleg meg tudta ugorni valamelyiket az alábbiak közül:

- $i-[(j-i)-1] \rightarrow i$
- $i-(j-i) \rightarrow i$
- $i-[(j-i)+1] \rightarrow i$

Kőszáli-kecske – 2

`a[0..n]` - a híd 0/1 konfigurációja

`c[0..n][0..n]` - `c[i][j]` tárolja, hogy megugorható-e $i \rightarrow j$ szakasz

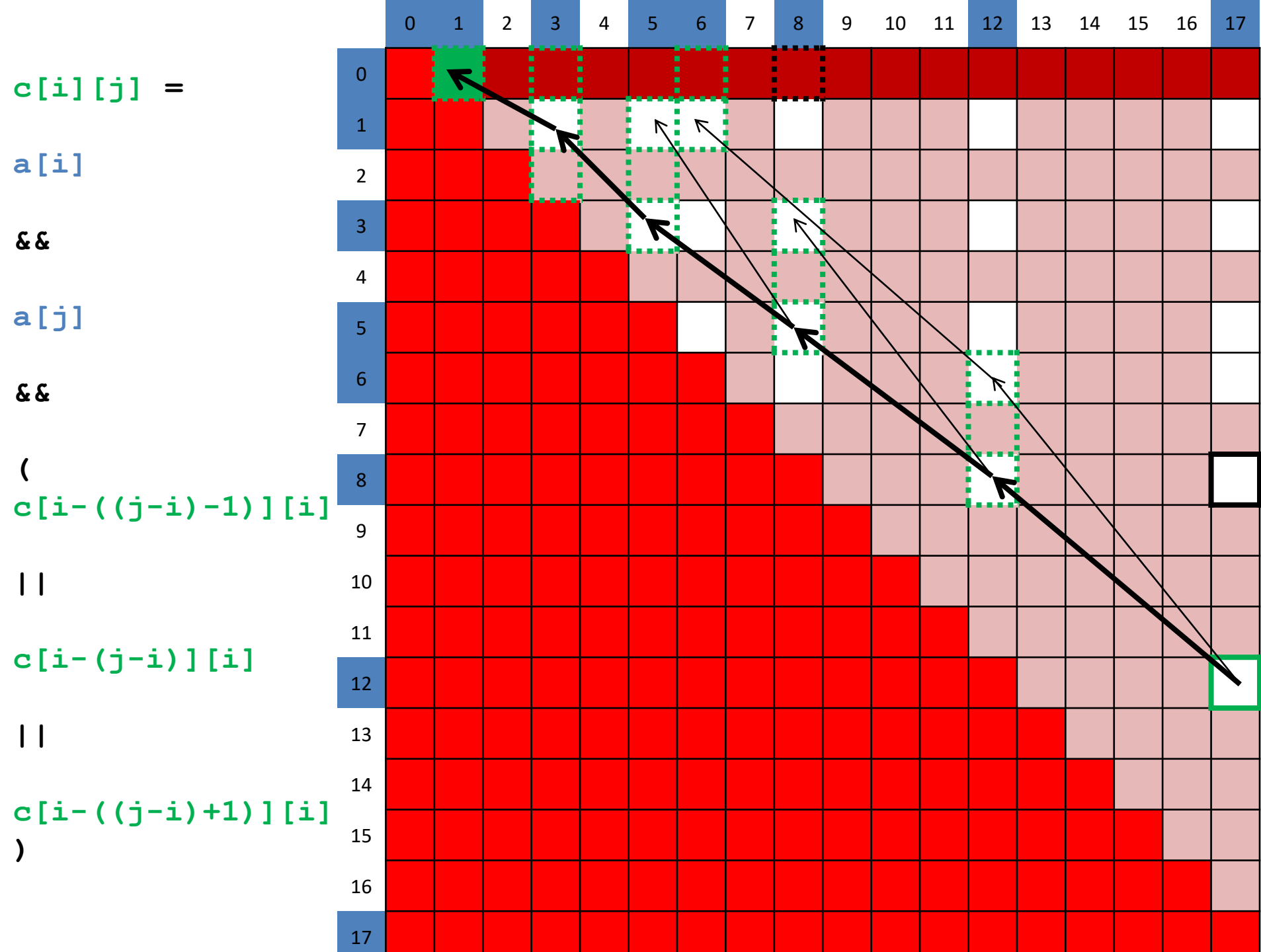
`c[i][j] = 0`, ha $i \geq j$

`c[0][1] = 1`

`c[0][j] = 0`, ha $j > 1$

`c[i][j] = a[i] && a[j] &&`

`(c[i - ((j - i) - 1)][i] || c[i - (j - i)][i] || c[i - ((j - i) + 1)][i])`



Kőszáli-kecske – 1 [GREEDY]

- Implementáld a Kőszáli-kecske-1 mohó algoritmust!
- Implementáld a Kőszáli-kecske-1 mohó algoritmust úgy, hogy nem tárolod el a híd konfigurációját!

Kőszáli-kecske – 2 [DP]

- n : hídhossz
- m : deszkaszám
- Implementáld a Kőszáli-kecske-2 dinamikus programozás algoritmust úgy, hogy $c[0..n][0..n]$ méretű tömböt használj!
- Implementáld a Kőszáli-kecske-2 dinamikus programozás algoritmust úgy, hogy $c[0..m][0..m]$ méretű tömböt használj!

A photograph of a mountain goat standing on a rocky peak. The goat is positioned on the right side of the frame, facing left. The rocks are dark and jagged, with some moss or lichen visible. The sky is overcast and grey. The overall scene is rugged and natural.

DINAMIKUS PROGRAMOZÁS

videó

Kőszáli kecske