

Algoritmi in podatkovne strukture



**Dinamično
programiranje in
spuščanje jajc**



Dinamično programiranje

- Ideja metode
 - več **prekrivajočih** se podnalog
 - že rešenih nalog ne rešujemo ponovno
 - shranjevanje rešitev podnalog
 - prihranek časa na račun prostora

Richard E. Bellman, 1920 – 1984



Dinamično programiranje

- Izvedba
 - od zgoraj navzdol
 - rešimo vse podnaloge in
 - si zapomnimo njihove rešitve (memoizacija)
 - od spodaj navzgor – klasičen pristop
 - začnemo pri majhnih nalogah in
 - postopoma gradimo rešitve večjih

Dva preprosta primera

- Fibonaccijeva števila

$$F_0 = 0$$

$$F_1 = 1$$

$$F_n = F_{n-2} + F_{n-1}$$



Dva preprosta primera

- Binomska števila

$$\binom{n}{0} = \binom{n}{n} = 1$$

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$



Dinamično programiranje

- Sestavine dinamičnega programiranja
 - prekrivanje podproblemov
 - neodvisnost podproblemov
 - rešitev enega podproblema ne vpliva na druge
 - optimalna podstruktura
 - optimizacijski problemi
 - optimalna rešitev vsebuje optimalne rešitve podproblemov

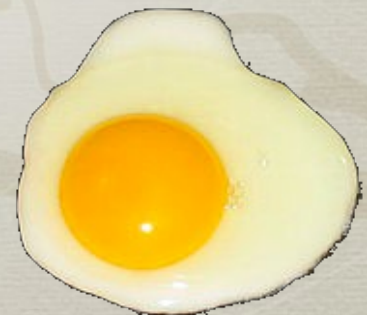
Spuščanje jajc

- Problem spuščanja jajc
 - Na voljo imate k popolnoma enakih jajc in eno n nadstropno stolpnico.
 - Katero je najvišje nadstropje, s katerega lahko še spustite jajce, da se ne razbije?
 - Minimiziramo število spuščanj.
 - Jajce lahko spustite kolikor krat hočete, da se le ne razbije.
 - Če se razbije pri metu iz nekega nadstropja, se tudi iz vseh višjih nadstropij.

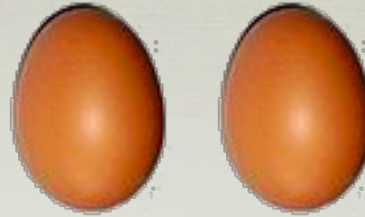
Spuščanje jajc



- Kaj, če imamo le eno jajce?
 - $k = 1$
- Kaj, če imamo neskončno jajc?
 - $k = \infty$
- Kdaj je jajc dovolj?



Spuščanje jajc



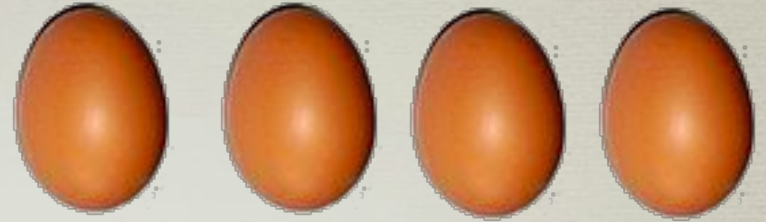
- Kaj, če imamo dve jajci?
 - $k = 2$

- Rešitev

- $x + (x-1) + (x-2) + \dots + 2 + 1 \geq n$

$$x = \left\lceil \frac{\sqrt{1+8*n}-1}{2} \right\rceil$$

Spuščanje jajc



- Kaj, če imamo več jajc?

- $s(n, k)$... najmanjše število spuščanj n nadstropij in k jajc

$$s(n, k) = 1 + \min_{1 \leq x \leq n} \max(s(x-1, k-1), s(n-x, k))$$

- $s(n, 1) = n$
- $s(0, k) = 0$
- $s(1, k) = 1$

