

Univerzitetni programerski maraton

FINALE 2019 – rešitve nalog

Tomaž Hočevan

Uradne rešitve

- najkrajša: 3 vrstic (Kazenske minute)
- najdaljša: 76 vrstic (Deutsch für Anfänger)
- povprečje: 35 vrstic
- skupaj: 317 vrstic
 - 1 vrstica/minuto ... :-)

Kazenske minute (28/28 zerodays @ 0:06)

Izračunaj kazenske minute vseh ekip skupaj.

- predpostavki:
 - po OK ni drugih oddaj
 - vsem uspe rešit nalogu
- a) preštej oddaje za pare (ekipa, naloga)
 - b) poišči nekaznovane oddaje

```
n = int(input())
s = {tuple(input().split()[:2]) for i in range(n)}
print((n-len(s))*20)
```

Češka (26/28 Final Solution 0:25)

Sabotiraj tekmovanje čim bolj v svojo korist.

	1	T	T+K	N
L				

- izračunaj točke vseh L ekip z “luknjo” od T do $T+K-1$
 - popravljaj točke iz T na $T+1$ v $O(L)$
 - vnaprej izračunaj predponske vsote
- izračunaj svojo uvrstitev (pazi na delitve mest)
 - uredi vse ekipe po točkah
 - preštej ekipe z večjim številom točk

Orientacija (24/27 Final Solution 0:39)

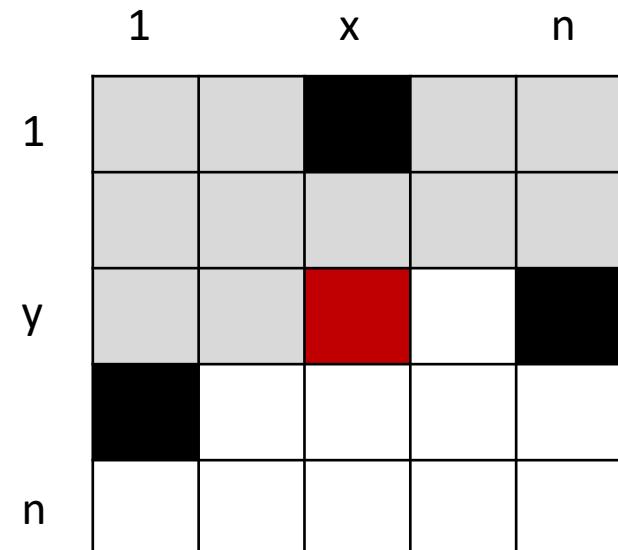
Izračunaj verjetnost poti po 2D polju.

- “dinamično programiranje”

a) poznamo $f(y, x)$

$$f(y+1, x) += p \cdot f(y, x)$$

$$f(y, x+1) += (1-p) \cdot f(y, x)$$



a) računamo $f(y, x)$

$$f(y, x) = f(y-1, x) \cdot p + f(y, x-1) \cdot (1-p)$$

Velodrom (13/15 Ctrl Alt Defeat 0:30)

Optimalno izvajaj menjave v kolesarski ekipi na velodromu.

13 10 6 10
5
-10 10 20 30
-3 1 2 3
-8 0 0 20
-5 10 10 10

krog	kolesarji	energija
1	1, 2, 3, 4	3, 11, 6, 20
2	2, 1, 3, 4	8, 13, 6, 30
3	2, 1, 3, 4	5, 23, 6, 40
4	2, 1, 3, 4	2, 33, 6, 50
5	1, 3, 4, 2	23, 6, 60, 5

- rekurzivno izčrpno preiskovanje
 - popravimo utrujenosti kolesarjev
 - izberemo novo lokacijo prvega kolesarja
 - $O(4^n)$

Moderno nogomet (6/16 ScoreOutOfBoundsException 0:28)

Razdeli igralce v dve čim bolj enakovredni ekipi.

1032 4533 3718 2323 3225 6817 721 9322 22 1995

$$|16848 - 16860| = 12$$

- Partition problem ... NP-hard
 - $N \leq 36$ brute force \times
 - $s_i \leq 10^9$ dinamično programiranje \times
- izberi ekipo, ki bo čim bližje $\frac{1}{2} \cdot \sum s_i$

Moderni nogomet

- Meet in the middle (split and merge)
 - vse podmnožice prvih in zadnjih $N/2$ elementov

$$24 = \{\underline{7}, \underline{1}, 3, \underline{2}, 6, \underline{5}\}$$

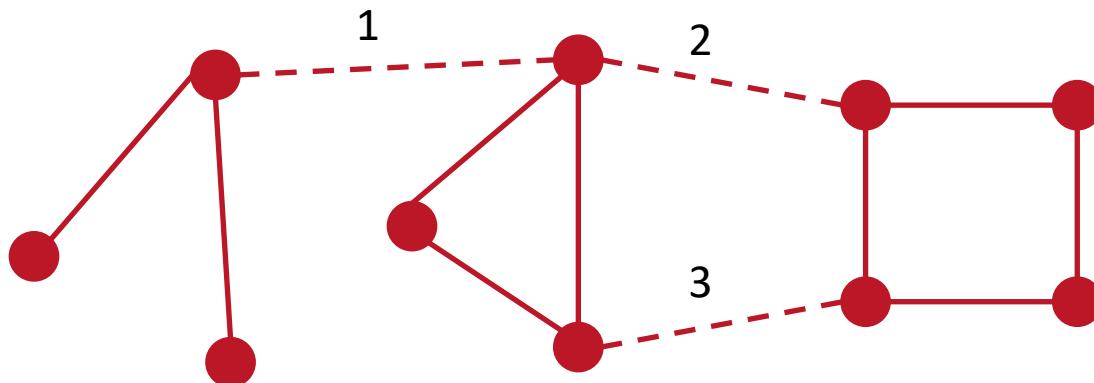
0 = {}
1 = {1}
3 = {3}
4 = {1, 3}
7 = {7}
8 = {7, 1}
10 = {7, 3}
11 = {7, 1, 3}

0 = {}
2 = {2}
5 = {5}
6 = {6}
7 = {2, 5}
8 = {2, 6}
11 = {6, 5}
13 = {2, 6, 5}

- $O(2^{N/2} \cdot N)$

Pajčevina (7/16 Final Solution 0:51)

Izračunaj število komponent po vsakem rezu povezave v grafu



- prerezemo most --> nova komponenta
- posodobimo mostove?

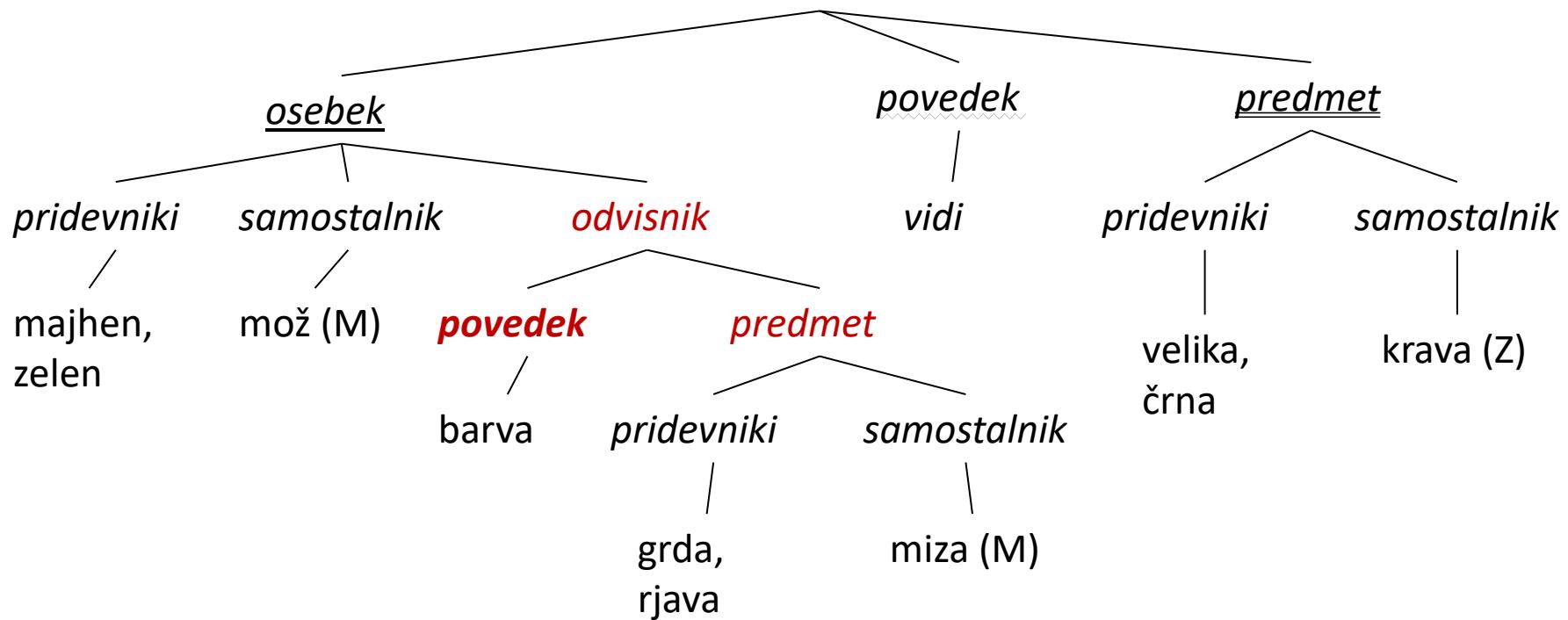
Pajčevina

- obrnemo proces in dodajamo povezave!
 - komponente = disjunktne množice vozlišč
 - dodajanje povezave = unija množic
-
- Disjoint-set / Union-find data structure

Deutsch für Anfänger (2/6 luftkanali 3:52)

Enostavne povedi v slovenščini prevedi v nemščino.

Majhen zelen mož, ki barva grdo rjava mizo, vidi veliko črno kravo.



Der kleine grüne Mann, welcher einen hässlichen braunen Tisch färbt, sieht eine grosse schwarze Kuh.

Neurejen seznam (1/3 Final Solution 3:09)

Preštej "neurejene" sezname.

- "neurejen" \neq vsi el. na napačnih mestih
 - npr. (8, 2, 2, 1)
 - kombinatorika, dinamično programiranje
- a) predpostavimo različne elemente oz. 1 ... n
- $$\begin{aligned} f(n) &= \text{št. neurejenih seznamov prvih } n \text{ elementov} \\ &= n! - (\text{seznami z vsej enim urejenim elementom}) \\ &= n! - \sum_i (\text{seznam s prvim urejenim el. na mestu } i) \end{aligned}$$

Neurejen seznam

- seznam s prvim “urejenim” el. na mestu i
 - pred njim so manjši, za njim pa večji
 - npr.: 3, 1, 2, **4**, 5, 7, 6, 8
 - $s[i] = i$
 - $s[1..i-1] = \text{neurejen seznam } \{1, \dots, i-1\}$
 - $s[i+1..n] = \text{permutacija } \{i+1, \dots, n\}$

$$f(n) = n! - \sum_i (f(i-1) \cdot (n-i)!)$$

Neurejen seznam

b) ponavljajoči elementi

$$f(n) = n! - \sum_i (f(i-1) \cdot (n-i)!)$$

$p(i, j)$ = permutacije elementov $x[i..j]$

$$f(n) = p(1, n) - \sum_i (f(i-1) \cdot p(i+1, n)!)$$

$$p(\{1,2,2,3,4,4,4\}) = 7! / (1! \cdot 2! \cdot 1! \cdot 3!)$$

$$p(\{1,2,2,3,4,4,4\}) = p(\{1,2,2,3,4,4\}) \cdot 7 / 3$$

- multiplikativni inverz za deljenje (mod = praštevilo)
 $\text{inv}(x) = x^{(\text{mod}-2)} \% \text{mod}$
- $O(n^2)$

Varianca (0/3)

Razdeli zaporedje na strnjene kose tako, da minimiziraš varianco vsot po kosih.

k=4	i	n
7	2	5
1	7	1
1	1	4
4	9	1
7	8	13
		10

$$\text{avg} = 1/K \sum_{i=1..K} v_i = 1/K \sum_{i=1..N} x_i$$

$$\text{var} = 1/K \sum_{i=1..K} (v_i - \text{avg})^2$$

$$f(k, n) = \min_{i < n} f(k-1, i) + \text{cost}(i+1, n)$$

čas: $O(n^2)$ ✗

Varianca

- monotonost delitvenih mest

$\text{opt}(k, n) = i$, ki minimizira $f(k, n)$

$\text{opt}(k, n) \leq \text{opt}(k, n+1)$

k=4	7	2	5	1	7	1	1	4	9	1
	7	2	5	1	7	1	1	4	9	
	7	2	5	1	7	1	1	4		
	7	2	5	1	7	1	1			
	7	2	5	1	7	1	1			

- * monotonost velja za vsako konveksno funkcijo $cost$, ki je odvisna od vsote skupine

Varianca

- divide and conquer optimization

`solve(k, nL, nR, iL, iR)`

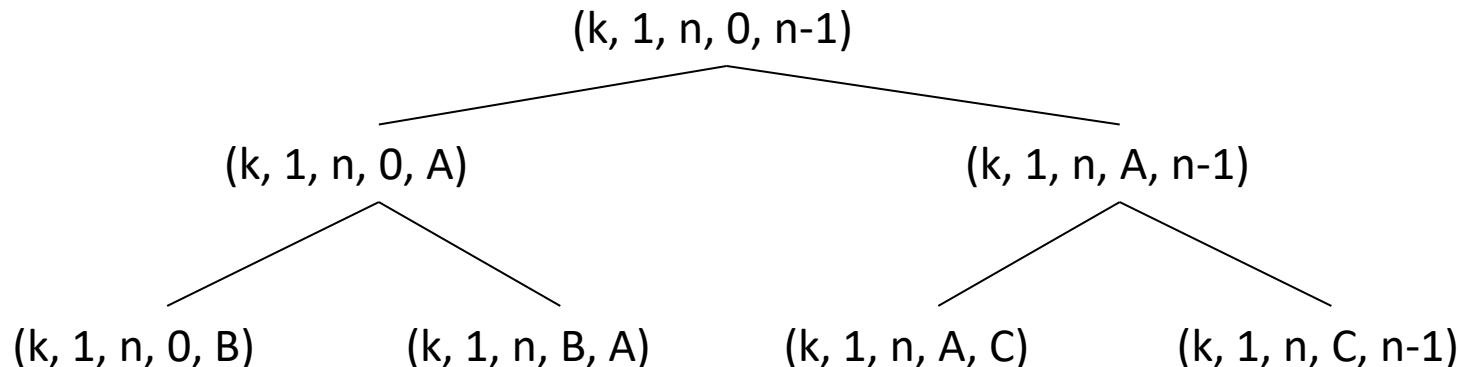
- rešimo za $n = (n_L + n_R)/2$
- izračunamo $f(k, n)$, $\text{opt}(k, n)$

$$\min_{i_L \leq i \leq i_R} [f(k-1, n) + ((c(n) - c(i)) - \text{avg})^2]$$

- `solve(k, nL, n-1, iL, opt(k,n))`
- `solve(k, n+1, nR, opt(k,n), iR)`

Varianca

- čas: $O(n \cdot \log n)$



$[0, n-1]$

$[0, A] + [A, n-1]$

$[0, B] + [B, A] + [A, C] + [C, n-1] \leq 2n$

Zaključek

- izognili ste se:
 - geometriji
 - računskim napakam
 - pretokom v grafih
 - algoritmom na nizih
 - drevesnim podatkovnim strukturam
 - ...
- “čudilo me bo, če ne bo imelo več ekip vse točke”