



Algoritmika

1. szeminárium



Szökőévek

Számítsuk ki hány szökőév volt/lesz két különböző évszám között!

Útmutatás: A szökőév osztható 4-gyel és nem osztható 100-zal, vagy osztható 400-zal.

Példák:

- 1964 szökőév
- 1900 nem szökőév
- 2000 szökőév

Szökőévek - naív algoritmus

- A két évszám között minden évről ellenőrizzük, hogy szökőév-e
- Egy változóban számoljuk a szökőévek számát.

Szökőévek - naív algoritmus

Algoritmus SzökőévekSzáma (év1, év2)

$SZ \leftarrow 0$

Minden év = év1, év2 **Végezd el**

Ha Szökőév(év) **akkor**

$SZ \leftarrow SZ + 1$

Vége (Ha)

Vége (Minden)

Szökőévek - gyorsítás

- Amint megtaláltuk az első szökőévet, utána elég, ha minden negyedik évről megvizsgáljuk, hogy szökőév-e

Szökőévek - gyorsítás

Algoritmus SzökőévekSzáma (év1, év2)

év \leftarrow év1, sz \leftarrow 0

Amíg év mod 4 \neq 0 **Végezd el**

év \leftarrow év + 1

Vége (Amíg)

Amíg év \leq év2 **Végezd el**

Ha Szökőév(év) **akkor**

sz \leftarrow sz+1

Vége (Ha)

év \leftarrow év + 4

Vége (Amíg)

Szökőévek - gyorsítás

- Számoljuk meg hány szökőév telt el 0 és év1 között (**x**)
- Számoljuk meg hány szökőév telt el 0 és év2 között (**y**)
- Az eredmény **y-x**

Konverzió

Adott egy szám a **p** számrendszerben.

Írjuk ki a szám **q** számrendszerbeli alakját. ($p, q \leq 10$)

Konverzió

Adott egy szám a **p** számrendszerben.

Írjuk ki a szám **q** számrendszerbeli alakját. ($p, q \leq 10$)

Példa:

Bemenet: $p=3, q=2$

$(\text{szám})_3 = 11012$

Kimenet:

$(\text{szám})_2 = 1110001$

Konverzió

Adott egy szám a **p** számrendszerben.

Írjuk ki a szám **q** számrendszerbeli alakját. ($p, q \leq 10$)

Példa:

Bemenet: $p=3, q=2$

$(\text{szám})_3 = 11012$

Kimenet:

$(\text{szám})_2 = 1110001$

Lépések:

- alakítsuk át a számot p számrendszerből 10-es számrendszerbe
- alakítsuk át a számot 10-es számrendszerből q számrendszerbe

Konverzió $p \rightarrow 10$

Algoritmus Konverzio $P \rightarrow 10$ (szám, p)

újszám $\leftarrow 0$

hatvány $\leftarrow 1$

Amíg szám $\neq 0$ **végezd el**

számjegy \leftarrow szám **mod** 10

szám \leftarrow szám **div** 10

újszám \leftarrow újszám + számjegy * hatvány

hatvány \leftarrow hatvány * p

Vége (Amíg)

Vissza újszám

Vége (Algoritmus)

Konverzió $10 \rightarrow q$

Algoritmus Konverzio $10 \rightarrow P$ (szam, p)

újszám $\leftarrow 0$

hatvány $\leftarrow 1$

Amíg szám $\neq 0$ **végezd el**

számjegy \leftarrow szám **mod** p

szám \leftarrow szám **div** p

újszám \leftarrow újszám + számjegy * hatvány

hatvány \leftarrow hatvány * 10

Vége (Amíg)

Vissza újszám

Vége (Algoritmus)

Konverzió $p \rightarrow q$

Amire figyeljünk:

- ha $p = 10$, akkor fölösleges a $p \rightarrow 10$ konverzió
- ha $q = 10$, akkor fölösleges a $10 \rightarrow q$ konverzió
- **Hibakezelés:** ellenőrizzük, hogy a kapott szám valóban p számrendszerben van (az összes számjegye kisebb mint p)

Konverzió $p \rightarrow q$

Algoritmus $P \rightarrow Q$ (szam, p, q)

Ha Ellenoriz(szam, p) **Akkor** { leellenőrizzük, hogy tényleg p számrendszerben van a szam }

Ha $p < 10$ **Akkor**

szam = Konverzio $P \rightarrow 10$ (szam, p)

Vége (Ha)

Ha $q < 10$ **Akkor**

szam = Konverzio $10 \rightarrow P$ (szam, q)

Vége (Ha)

Vége (Ha)

Vissza szám

Vége (Algoritmus)

Konverzió $p \rightarrow q$

Az algoritmus még mindig nem teljes:

- Fennáll a túlcsoordulás lehetősége