Pēteris Račinskis
pr20015

Mājas darbs 1

1. Eilera funkcija

Parādīt sakarību:

1. $\forall d|n : A_n(d) = \{k \in [n] \mid (n,k) = d\}$
2. $\sum_{d|n} |A_n(d)| = n$
3. $k \in A(d) \to k = d\lambda \to$
   $\to \lambda \in [\frac{n}{d}]; \lambda \neq n$, ja $\frac{n}{d} > 1; \lambda \perp \frac{n}{d}$, citādi $(n,k) > d$; $\to$
   $\to |A_n(d)| = |\{x \in [\frac{n}{d}] \mid x \perp \frac{n}{d}\}| = \varphi(\frac{n}{d})$
4. $\{\frac{n}{d} \in [n] \mid \frac{n}{d}|n\} \leftrightarrow \{d \in [n] \mid d|n\} \to$
   $\to \sum_{d|n} \varphi(d) = \sum_{d|n} \varphi(\frac{n}{d}) = \sum_{d|n} |A_n(d)| = n$

Izmantot inversiju:

1. $g(n) = n; f(d) = \varphi(d)$
2. $n = \sum_{d|n} \varphi(d) \leftrightarrow \varphi(n) = \sum_{d|n} \mu(d)\frac{n}{d} = n\sum_{d|n} \frac{\mu(d)}{d}$
3. Izplešot reizinājumu:
   $\prod_{p|n}(\frac{1}{1-p}) = 1 - \frac{1}{p_1} - \frac{1}{p_2} - ... - \frac{1}{p_k} + \frac{1}{p_1 p_2} + ... \frac{1}{p_{k-1}p_k} - ....$
   t.i., pirmskaitļi saucējā grupējas nepāra daļās ar - un pāra daļās ar + zīmi. Vienā reizinātājā tas pats pirmskaitlis neatkārtojas.
4. $d = p_1^{a_1} p_2^{a_2} ... p_k^{a_k} : a_i > 0; \exists a_i > 1 \to \mu(d) = 0 \vee \mu(d) = (-1)^k$
   tātad summā $\sum_{d|n} \frac{\mu(d)}{d}$ ar nenulles skaitītāju parādās tikai tādi paši saskaitāmie, kā augstāk dotajā reizinājumā, un pārējo zīmes sakrīt. Līdz ar to:
5. $n\sum_{d|n} \frac{\mu(d)}{d} = n\prod_{p|n}(\frac{1}{1-p})$

2. *Mobiusa funkcija daļēji sakārtotai kopai*

Inversā funkcija konvolūcijai:

- $f^{-1}(x,x) = \dfrac{1}{f(x,x)}$, jo $f(x,x) \cdot \dfrac{1}{f(x,x)} = 1 = \delta(x,x)$
- $f^{-1}(x,y) = -\dfrac{1}{f(y,y)} \sum\limits_{z: x \leq z < y} f^{-1}(x,z) f(z,y)$, jo

$$(f * f^{-1})(x,y) = \sum_{z: x \leq z < y} f^{-1}(x,z) f(z,y) - \frac{f(y,y)}{f(y,y)} \sum_{z: x \leq z < y} f^{-1}(x,z) f(z,y) =$$
$$= 0 = \delta(x,y), x \neq y$$

Tā kā $\mu(x,y) = \zeta^{-1}(x,y)$ un $\forall x \leq y; \zeta(x,y) = 1$, izteiksme vienkāršojas uz sekojošo:

- $\mu(x,x) = 1$
- $\mu(x,y) = -\sum\limits_{z: x \leq z < y} \mu(x,z)$

Pēc šīs metodes var izrēķinat funkcijas vērtības un saglabāt tās matricā. Vieglāt to ir darīt ar datoru - tad var arī pārliecināties, ka iegūtā funkcija patiešām ir inversā - ja funkcijas uzdotas kā n*n matricas, konvolūcijas izteiksmi var pārrakstīt kā $(\zeta * \mu)(a,b) = \begin{bmatrix} \zeta_{a1} & \dots & \zeta_{an} \end{bmatrix} \begin{bmatrix} \mu_{1b} \\ \dots \\ \mu_{nb} \end{bmatrix}$, bet rezultātam jābūt Kronekera deltai (vienības matricai).

```
[peter@peter-laptop-old:~/repos/homework/KMB_1]$ ./compute-mobius.py 1 2
zeta(i,j)
[1      1       1       1       1       1]
[0      1       1       0       0       1]
[0      0       1       0       0       1]
[0      0       0       1       0       1]
[0      0       0       0       1       1]
[0      0       0       0       0       1]
mu(i,j)
[1      -1      0       -1      -1      2]
[0      1       -1      0       0       0]
[0      0       1       0       0       -1]
[0      0       0       1       0       -1]
[0      0       0       0       1       -1]
[0      0       0       0       0       1]
(zeta*mu)(i,j)
[1      0       0       0       0       0]
[0      1       0       0       0       0]
[0      0       1       0       0       0]
[0      0       0       1       0       0]
[0      0       0       0       1       0]
[0      0       0       0       0       1]
[1,2] -1
```

Pēteris Račinskis
pr20015

3. Programma Mobiusa funkcijas aprēķinam

Programma python (3.8+) valodā iekļauta pielikumā kopā ar instrukciju un trim paraugiem - jau doto daļēji sakārtoto kopu, galīgu naturālo skaitļu apakškopu un skaitļa 30 dalītājiem. Noņemot komentāra simbolus iespējams nomainīt ieejas faila nosakumu un iestatīt zeta, mobius un sarēķinātās konvolūcijas matricu izvadi uz konsoles (pēc noklusējuma matricas izvadītas netiek).

Izvade naturāliem skaitļiem līdz 8:

```
[peter@peter-laptop-old:~/repos/homework/KMB_1]$ ./compute-mobius.py 1 2
zeta(i,j)
[1      1       1       1       1       1       1       1]
[0      1       1       1       1       1       1       1]
[0      0       1       1       1       1       1       1]
[0      0       0       1       1       1       1       1]
[0      0       0       0       1       1       1       1]
[0      0       0       0       0       1       1       1]
[0      0       0       0       0       0       1       1]
[0      0       0       0       0       0       0       1]
mu(i,j)
[1      -1      0       0       0       0       0       0]
[0      1       -1      0       0       0       0       0]
[0      0       1       -1      0       0       0       0]
[0      0       0       1       -1      0       0       0]
[0      0       0       0       1       -1      0       0]
[0      0       0       0       0       1       -1      0]
[0      0       0       0       0       0       1       -1]
[0      0       0       0       0       0       0       1]
(zeta*mu)(i,j)
[1      0       0       0       0       0       0       0]
[0      1       0       0       0       0       0       0]
[0      0       1       0       0       0       0       0]
[0      0       0       1       0       0       0       0]
[0      0       0       0       1       0       0       0]
[0      0       0       0       0       1       0       0]
[0      0       0       0       0       0       1       0]
[0      0       0       0       0       0       0       1]
[1,2] -1
```

Izvade skaitļa 30 dalītājiem:

```
[peter@peter-laptop-old:~/repos/homework/KMB_1]$ ./compute-mobius.py 1 2
zeta(i,j)
[1      1       1       1       1       1       1       1]
[0      1       0       0       1       1       0       1]
[0      0       1       0       1       0       1       1]
[0      0       0       1       0       1       1       1]
[0      0       0       0       1       0       0       1]
[0      0       0       0       0       1       0       1]
[0      0       0       0       0       0       1       1]
[0      0       0       0       0       0       0       1]
mu(i,j)
[1      -1      -1      -1      1       1       1       -1]
[0      1       0       0       -1      -1      0       1]
[0      0       1       0       -1      0       -1      1]
[0      0       0       1       0       -1      -1      1]
[0      0       0       0       1       0       0       -1]
[0      0       0       0       0       1       0       -1]
[0      0       0       0       0       0       1       -1]
[0      0       0       0       0       0       0       1]
(zeta*mu)(i,j)
[1      0       0       0       0       0       0       0]
[0      1       0       0       0       0       0       0]
[0      0       1       0       0       0       0       0]
[0      0       0       1       0       0       0       0]
[0      0       0       0       1       0       0       0]
[0      0       0       0       0       1       0       0]
[0      0       0       0       0       0       1       0]
[0      0       0       0       0       0       0       1]
[1,2] -1
```

Pēteris Račinskis
pr20015

4. Mobiusa funkcija naturāliem skaitļiem pēc <=

Pēc programmas izvades jau uzreiz redzams, funkcija pieņem vērtības

$\mu(x, x) = 1$

$\mu(x, y) = -1, y = x + 1$

$\mu(x, y) = 0, y \neq x \wedge y \neq x + 1$

To var konstatēt arī apskatot 2. uzdevumā doto formulu - pēc pirmajiem diviem saskaitāmajiem summā vairs nevienam citam intervālam nav nekādas nozīmes, jo iepriekšējo summa ir 0.

Tātad inversija ir:

$$f(t) = \sum_{s \leq t} g(s) \leftrightarrow g(t) = \sum_{s \leq t} f(s)\mu(s, t)$$

kas sakrīt ar intuīciju, ka

$$f(g) = g(1) + g(2) + ... + g(t) \leftrightarrow g(t) = f(t) - f(t-1)$$

7. (tikai nedaudz iesākts uzdevums) Pāru sakārtojums

Uzdevums nav pabeigts, jo neizdevās atrast metodi, kā aprēķināt r_k^G, taču princips vienam potenciālam risinājumam ir izdomāts.

Pieņemot, ka pie galda ir n pozīcijas pāriem (2n, ja vīriešu un sieviešu vietas pabīda pa vienu pozīciju, ko var piereizināt klāt pēc tam), i-tā pāra otrais partneris nedrīkst ieņemt pozīcijas i = j un i + 1 mod n = j (jo no vienas puses tāpat sanāktu sēdēt blakus). Iegūst matricu ar šādu aizliegto pozīciju galdiņu:

$$\begin{bmatrix} X & X & . & . & .... & . & . \\ . & X & X & . & ... & . & . \\ . & . & X & X & ... & . & . \\ ... & & & & & & \\ . & . & . & . & ... & X & X \\ X & . & . & . & ... & . & X \end{bmatrix}$$

Tad attiecīgi varētu aprēķināt

$$N_{0,n}^G = \sum_{k=0}^{n} (-1)^k (n-k)! r_k^G$$

, taču autors nav paspējis izdomāt, kā pareizi rēķināt r_k^G (ja tā vispār ir pareizā pieeja, varbūt problēmu vajadzēja apskatīt no citas puses).