Prepisovalni sistemi

Imejmo operacije f_i mestnosti d_i nad množico objektov A. **Konkretna predstavitev** te množice uporablja neko končno abecedo Σ in odločljivo množico $T \subseteq \Sigma^*$ ter izračunljive operacije g_i enakih mestnosti d_i. Poleg tega imamo vrednostno funkcijo $v\colon T \to A$, tako da je v surjektivna in da $v(g_i(t_1, \ldots, t_{d_r})) = f_i(v(t_1), \ldots, v(t_r))$, oz. da komutira diagram

$$\begin{array}{ccc} T^{d_i} & \xrightarrow{g_i} & T \\ {\scriptstyle v^{d_i}}\downarrow & & \downarrow{\scriptstyle v} \\ A^{d_i} & \xrightarrow{f_i} & A\,. \end{array}$$

Ekvivalenca termov: Za dva terma $t_1, t_2 \in T$ definiramo $t_1 \sim t_2 \iff v(t_1) = v(t_2)$. Struktura (A, f_i) je izračunljiva, če obstaja poleg pogojev zgoraj tudi tak izračunljiv epimorfizem v.

Kanonska funkcija: izbere po enega predstavnika iz vsakega ekvivalenčnega razreda. Funkcija f je kanonska za $(T, \sim)$, če $f(t) \sim t$, $t_1 \sim t_2 \implies f(t_1) = f(t_2)$, f izračunljiva.

Redukcijska relacija: Relacija $R \subseteq T \times T$.

oznaka	pomen	razlaga
$\to$	R	redukcijska relacija
$\overset{n}{\to}$	R^n	v n korakih
$\leftarrow$	R^{-1}	inverz redukcijske relacije
$\leftrightarrow$	$R \cup R^{-1}$	simetrična ovojnica
$\overset{+}{\to}$	R^+	tranzitivna ovojnica
$\overset{*}{\to}$	R^*	relf. in tranzitivna ovojnica
$\overset{*}{\leftrightarrow}$	$(R \cup R^{-1})^*$	ekvivalenčna ovojnica

Za dva elementa definiramo **imata skupnega naslednika**: $a \downarrow b \iff a \overset{*}{\to} c \overset{*}{\leftarrow} b$.
Za element definiramo, da je **reduciran**: $a \downarrow \iff \neg\exists b \in T\colon a \to b$.
Prepisovalni sistem: je par $(T, \to)$, v katerem za $\sim$ vzamemo $\overset{*}{\leftrightarrow}$.

Relacija $\to$ je **Noetherska**, če ne vsebuje neskončnih redukcijskih verig.
Trd: $\to$ Noetherska $\iff \exists f\colon T \to \mathbb{N}\ \forall t_1, t_2 \in T\colon t_1 \to t_2 \implies f(t_1) > f(t_2)$
Trd: $\to$ Noetherska $\iff \forall t \in T\ \exists r \in T\colon t \overset{*}{\to} r \downarrow$.

Def: Relacija $\to$ ima **enolične reducirane oblike** (ERO), če $\forall a, b, c \in T\colon (\downarrow b \overset{*}{\leftarrow} a \overset{*}{\to} c \downarrow) \implies b = c$.
Relacija $\to$ je **konfluentna** (KON), če $\forall a, b, c \in T\colon (b \overset{*}{\leftarrow} a \overset{*}{\to} c) \implies b \downarrow c$.
Relacija $\to$ je **lokalno konfluentna** (LKON), če $\forall a, b, c \in T\colon (b \leftarrow a \to c) \implies b \downarrow c$.
Relacija $\to$ ima **Church-Rosserjevo** lastnost (CR), če $\forall a, b \in T\colon (a \overset{*}{\leftrightarrow} b) \implies a \downarrow b$.

Velja: CR $\implies$ KON $\implies$ LKON in KON $\implies$ ERO.
Newmanova lema: Če je $\to$ Noetherska, so KON, LKON, ERO in CR ekvivalentne.

Def: $\to$ je **polna**, če je Noetherska in konfluentna. Problem napolnitve relacije: radi bi razširili $\to$ tako, da bo polna, relacija $\overset{*}{\leftrightarrow}$ pa se bo ohranila.
Izrek: Za vsako Noethersko $\to$ v T obstaja napolnitev. Poiščemo jo tako, da najdemo izraz, ki ima dve ereducirani obliki in dodamo v relacijo pravilo, ki eno prevede na drugo.

To naprej (kritični pari) je nagravžno in upamo da ne bo na izpitu.

Hint: Kadar dokazujemo, da je zaporedje Noethersko, si lahko pogosto pomagamo s kakšno funkcijo (glej trditev pod definicijo Noetherskosti). Ker pogosto delamo na nizih števk, lahko pogosto uporabimo kar samo vrednost števila, ki nam ga zapis predstavlja (recimo za nize 0 in 1 uporabimo funkcijo, ki vrednoti dvojiški zapis). Pogosto moramo na začetek niza postaviti še 1 (0 in 000 oba kodirata 0, z dodajanjem pa dobimo 1 0 in 1 000, ki kodirata različni števili).
Ko testiramo enolične reducirane oblike si lahko pogosto pomagamo z ugotovitvami narave 'vsa pravila ohranjajo sodost oz. lihost števila ničel'. V nekaterih primerih, moramo gledati po kakšnem modulu.
Če imajo pravila med sabo 'prazen presek' (torej delujejo na disjunktnih podintervalih, recimo 00

in 11), iz tega sledi LKON. Če primerjamo za P_1 in P_2, nam redukcija P_1 in nato P_2 reducira v isti element kot P_2 in nato P_1, torej imata redukciji z P_1 oz. P_2 skupnega naslednika.

Polinomska aritmetika

Vedno naj bo K komutativen kolobar z $1 \neq 0$.

Rezultanta: Naj bosta $p(x) = \sum_{i=0}^{n} a_i x^i$ in $q(x) = \sum_{i=0}^{m} b_i x^i$ polinoma iz $K[x]$, $a_n \neq 0$, $b_m \neq 0$, $\operatorname{st} p + \operatorname{st} q > 0$ (to velja skos). Potem je rezultanta determinanta te $(n+m) \times (n+m)$ matrike

$$\operatorname{Rez}(p,q) = \det \begin{bmatrix} a_n & a_{n-1} & a_{n-2} & \cdots & 0 & 0 & 0 \\ 0 & a_n & a_{n-1} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_1 & a_0 & 0 \\ 0 & 0 & 0 & \cdots & a_2 & a_1 & a_0 \\ b_m & b_{m-1} & b_{m-2} & \cdots & 0 & 0 & 0 \\ 0 & b_m & b_{m-1} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & b_1 & b_0 & 0 \\ 0 & 0 & 0 & \cdots & b_2 & b_1 & b_0 \end{bmatrix}.$$

Trd: $\operatorname{Rez}(q,p) = (-1)^{mn} Rez(p,q)$
Trd: $n = 0 \implies Rez(p,q) = a_0^m$
Trd: $\exists r, s \in K[x], r \neq 0, s \neq 0$, tako da $\operatorname{st} r < \operatorname{st} q, \operatorname{st} s < \operatorname{st} p$ in $\operatorname{Rez}(p,q) = pr + qs$
Izr: Če K Gaussov ali cel: p in q imata nekonstanten skupni faktor $\iff \operatorname{Rez}(p,q) = 0$.
Trd: K cel, $p, q, r \in K[x]$, $\operatorname{st} p + \operatorname{st} r > 0$, $\operatorname{st} q + \operatorname{st} r > 0$, potem velja $\operatorname{Rez}(pq, r) = \operatorname{Rez}(p,r) \operatorname{Rez}(q,r)$ in $\operatorname{Rez}(r, pq) = \operatorname{Rez}(r,p) \operatorname{Rez}(r,q)$.
Trd: K cel, $p(x) = a \prod (x - \alpha_i), q(x) = b \prod (x - \beta_j)$.
Potem je $\operatorname{Rez}(p,q) = a^m \prod_i q(\alpha_i) = b^n \prod_j p(\beta_j) = a^m b^n \prod_i \prod_j (\alpha_i - \beta_j)$.
Lema o homomorfizmu: K, K_1 cela, $\varphi\colon K \to K_1$ homomorfizem, $\varphi(\operatorname{vk}(p)) \neq 0$. Potem $\varphi(\operatorname{Rez}(p,q)) = \varphi(\operatorname{vk}(p))^{\operatorname{st} q - \operatorname{st} \varphi(q)} \operatorname{Rez}(\varphi(p), \varphi(q))$ in $\varphi(\operatorname{Rez}(p,q)) = 0 \iff \operatorname{Rez}(\varphi(p), \varphi(q)) = 0$.

Uporaba: eliminacija spremenljivk
Izr: K Gaussov, $p, q \in K[x_1, ..., x_d], R = \operatorname{Rez}_{x_d}(p,q) \in K[x_1, \ldots x_{d-1}]$, F alg. zaprtje ulomkov K. Potem lahko spremenljivko eliminiramo
$\forall c \in F^d\colon p(c_1, \ldots, c_d) = q(c_1, \ldots, c_d) \implies R(c_1, \ldots, c_{d-1}) = 0$ in
in rešitev prenesemo nazaj: $\forall c \in F^{d-1}\colon R(c_1, \ldots, c_{d-1}) = 0 \neq \operatorname{vk}_{x_d}(p)(c_1, \ldots, c_{d-1}) \implies \exists c_d \in F\colon p(c_1, \ldots, c_d) = q(c_1, \ldots, c_d) = 0$
Tako lahko namesto enačb $p(x_1, \ldots, x_d) = 0$ in $q(x_1, \ldots, x_d) = 0$ pišemo enao samo enačbo $\operatorname{Rez}_{x_d}(p,q) = 0$. Pri tem se lahko pojavijo parazitske rešitve, če $\operatorname{vk}_{x_d}(p)(c_1, \ldots, c_{d-1}) = 0$.

Uporaba: če imamo algebraični števili a in b, potem s pomočjo rezultante dobimo polinome, ki uničijo $a + b$, ab, $1/a$, a^r, $r \in \mathbb{Q}^+$. Za vsako elgebraično število a obstaja polinom z racionalnimi koeficienti (in z celimi), ki ga uniči. Takemu z minimalno stopnjo in vodilnim koeficientom 1, se reče **minimalni** polinom. Vsak drug polinom, ki uniči a, je deljiv z minimalnim.

Algoritem 1 Evklidov algoritem (tudi za polinome)
Vhod: $a, b \in K$.
Izhod: Največji skupni delitelj $\gcd(a,b) \in K$

```
1: procedure GCD(a, b)
2:     while b ≠ 0 do
3:         r ← a mod b
4:         a ← b
5:         b ← r
6:     end while
7:     return a
8: end procedure
```

Razstavljanje polinomov

K ima karakteristiko 0. Polinom $f \in K[x]$ želimo razstaviti na $f = g \circ h$, $g, h \in K[x]$ in $1 < \operatorname{st} g, \operatorname{st} h < \operatorname{st} f$.

Trd: Če je $\operatorname{st} f$ praštevilo, razstavitev ne obstaja.
Poljubno si lahko izberemo prosti člen (sicer: $g_1(x) = g(x-c), h_1(x) = h(x)+c$) in vodilni koeficient h (sicer: $g_2(x) = g(x/c), h_2(x) = ch(x)$), izberemo $h(0) = 0$ in h moničen. Glej algoritem 2.

Algoritem 2 Razstavljanje polinomov (Kozen, Landau).
Vhod: $f \in K[x]$, $f = x^{rs} + a_{rs-1}x^{rs-1} + \cdots + a_0$, $r, s \geq 2$.
Izhod: $h, g \in K[x]$, tako da $f = g \circ h$, $g = x^r + b_{r-1}x^{r-1} + \cdots + b_0$, $h = x^s + c_{s-1}x^{s-1} + \cdots + c_0$, ali "Ni mogoče.".

---

procedure RAZSTAVI(f, r, s)
  $h_0 \leftarrow x^s$ ▷ Izračun h.
  for $k \leftarrow 1$ **to** $s-1$ **do**
    $h_k \leftarrow h_{k-1} + \frac{1}{r}(a_{rs-k} - [x^{rs-k}]h_{k-1}^r)x^{s-k}$
  end for
  $h \leftarrow h_{s-1}$
  $b_r \leftarrow 1$ ▷ Izračun g.
  for $i \leftarrow r-1$ **to** 1 **do** ▷ Gaussova eliminacija za $Ub = a$.
    $b_i \leftarrow a_{is} - \sum_{j=i+1}^{r}([x^{is}]h^j(x))b_j$
  end for
  $b_0 \leftarrow a_0$
  $g(x) \leftarrow \sum_{i=0}^{r} b_i x^i$
  if $f(x) = g(h(x))$ **then**
    return (g, h) ▷ Preverimo, če se izide.
  else
    return "Ni mogoče."
  end if
end procedure

Razcep polinomov v $\mathbb{Z}_p$, p praštevilo

Lastnosti $\mathbb{Z}_p$: $a^{p-1} = 1$, $(a+b)^p = a^p + b^p$, $v(x^p) = v(x)^p$, $u' = 0 \iff \exists v \in \mathbb{Z}[x] \colon v(x^p) = u(x)$.

Želimo razcepiti u. Izračunamo $d = \gcd(u, u')$. Če je $0 < \operatorname{st} d < \operatorname{st} u$, potem je $u = d \cdot u/d$ netrivialen razcep, ki ga lahko razcepimo naprej. Če je $\operatorname{st} d = \operatorname{st} u > \operatorname{st} u'$, potem $d | u'$ torej $u'(x) = 0$ in $u(x) = v(x^p) = v(x)^p$ in razcepimo v naprej. Če je $\operatorname{st} d = 0$, potem je u brez kvadratov in uporabimo algoritem 3.

Algoritem 3 Razcep polinomov (Berlekamp).
Vhod: $u \in \mathbb{Z}_p[x]$ brez kvadratov, $n = \operatorname{st} u \geq 2$.
Izhod: Netrivialen razcep polinoma u, če obstaja.

---

procedure RAZCEPI(u, p)
  Izračunaj $Q \in \mathbb{Z}_p^{n \times n}$ s koeficienti danimi z $x^{kp} = \sum_{i=0}^{n-1} q_{ik}x^k \pmod{u(x)}$.
  Poišči bazo $\{v^{(1)} = [1,0,\ldots,0] \cong 1 + 0x + \cdots, v^{(2)}, \ldots, v^{(r)}\}$ jedra $Q - I$ nad $\mathbb{Z}_p$.
  if $r = 1$ **then**
    return u nerazcepen.
  else
    for $a \leftarrow 0$ **to** $p-1$ **do**
      $d_a(x) = \gcd(u(x), v^{(2)}(x) - a)$
      if $\operatorname{st} d_a > 0$ **then**
        return $(d_a(x), u(x)/d_a(x))$ je razcep. ▷ Če želimo vseh r faktorjev, potem ne
      end if ▷ vrnemo takoj, ampak nadaljujemo
    end for ▷ do $p-1$ in nato naprej z $v^{(3)}$ itd.,
  end if ▷ dokler jih nimamo r.
end procedure

Razcep polinomov nad $\mathbb{Z}$

Def: Polinom $p = \sum_i a_i x^i$ je primitiven, če je $\gcd(a_0, \ldots, a_n) = 1$.
Lema: Produkt primitivnih je primitiven.

Ideja 1: razcepimo polinom po modulu p, kjer je p pračtevilo večje od $2M$ in M je tak, da so vsi koeficienti v razcepu po abs. manjši od M (obstajajo ocene).

Ideja 2: Razcep po nekem manjšem modulu p lahko dvignemo do razcepa nad $p^k < M$. Tukaj uporabimo Henslov dvig. Ko imamo faktorje nad Z_{p^k}. Potem gremo čez vse podmnožice faktorjev in poskusimo, če kakšen produkt deli p.

Rešetke

Celoštevilska ogrinjača vektorjev $v_1, \ldots v_k \in \mathbb{R}^m$ je množica $L(v_1, \ldots, v_k) = \{\sum \lambda_i v_i; \lambda_i \in \mathbb{Z}\}$.
Množica $\Lambda \subseteq \mathbb{R}^m$ je n-razsežna **rešetka**, če obstajajo $b_1, \ldots, b_n \in \mathbb{R}^m$, tako da $\Lambda = L(b_1, \ldots, b_n)$.
Množica $\{b_i\}$ je baza Λ, $m \times n$ matrika B s stolpci b_i pa bazna matrika.
Dve rešetki Λ_1, Λ_2 dim. n sta **enaki** $\iff$ obstaja $U \in \mathbb{Z}^{n \times n}$, da $B_2 = B_1 U$ in $\det U = \pm 1$.
Determinanta rešetke Λ z bazo B je $d(\Lambda) = \sqrt{\det(B^\mathsf{T} B)}$.
Pravokotnost baze meri $\delta(b_1, \ldots, b_n) = \prod \|b_i\| / d(\Lambda)$. Vedno je $\delta(\Lambda) \geq 1$.
Lema: Vsaka omejena podmnožica rešetke je končna.

SVP je problem iskanja najkrajšega vektorja v rešetki. SVP v 2D je iskanje najbližjega vektorja pravokotni projekciji: $(u, v) \to (u, v - ku)$, optimalni k je $\left\lceil \frac{\langle u,v \rangle}{\langle v,v \rangle} \right\rfloor$, kjer je $[x]$ najbližje celo število x. Od tod sledi algoritem 4. Posplošitev je algoritem 5.

Algoritem 4 Algoritem 60° za iskanje najkrajšega vektorja v rešetki dimenzije 2.
Vhod: Baza $(u, v) \in \mathbb{R}^m$.
Izhod: Nova baza (u, v), da je $\cos \measuredangle(u, v) \leq \frac{1}{2}$. Krajši izmed obeh je najkrajši vektor v rešetki.

```
1: procedure 60DEG(u, v)
2:     repeat
3:         swap(u, v)
4:         k ← ⌈⟨u,v⟩ / ⟨v,v⟩⌋
5:         u ← u − kv.
6:     until ‖v‖² ≤ ‖u‖²
7:     return (u, v).
8: end procedure
```

Algoritem 5 Algoritem ϑ°.
Vhod: Baza $(u, v) \in \mathbb{R}^m$ in $t \in [1, 2)$.
Izhod: Nova baza (u, v), da je $\cos \measuredangle(u, v) \leq \frac{t}{2}$.

```
1: procedure THETADEG(u, v, t)
2:     repeat
3:         swap(u, v)
4:         k ← ⌈⟨u,v⟩ / ⟨v,v⟩⌋
5:         u ← u − kv.
6:     until ‖v‖² ≤ t²‖u‖²
7:     return (u, v).
8: end procedure
```

Def: $(b_1, \ldots, b_n)$ baza rešetke Λ. Za **Gram-Schmidtovo bazo** $(b_1^*, \ldots, b_n^*)$ velja $b_i^* \perp b_j^*$ (za $i \neq j$) in $\mathcal{L}(b_1, \ldots, b_i) = \mathcal{L}(b_1^*, \ldots, b_i^*)$ za $i = 1, \ldots, n$.
Trd: $(b_1, \ldots, b_n)$ baza rešetke Λ potem je $\forall v \in \Lambda$: $\min_{1 \leq i \leq n} \|b_i^*\| \leq \|v\|$.
Def: $(b_1, \ldots, b_n)$ baza rešetke Λ potem je $b_j(i)$ **pravokotna projekcija vektorja** b_j vzoldž $\mathcal{L}(b_1^*, \ldots, b_i^*)$ na $\mathcal{L}(b_i, \ldots, b_n)$.
Def: Baza rešetke je ***t*-reducirana**, če velja $|\mu_{ij}| \leq \frac{1}{2}$ in $\|b_i^*\|^2 \leq t\|b_{i+1}(i)\|^2$ za $1 \leq i \leq n$.
Trd: Baza, ki jo vrne algoritem 5 je t-reducirana. Za splošno dimenzionalno rešetko uporabljamo algoritem 6.

Algoritem 6 Algoritem Lenstra–Lenstra–Lovász.
Vhod: Baza $b_1, \ldots, b_n \in \mathbb{R}^m$ rešetke Λ in $t \in [1, 2)$.
Izhod: t-reducirana baza $(b_1, \ldots, b_n)$ rešetke Λ

procedure LLL($\{b_i\}, t$)
    $\mu_{i,k} := \frac{\langle b_i, b_k^* \rangle}{\|b_k^*\|^2}$
    $k \leftarrow 1$
    while $k < n$ **do**
        $b_{k+1} \leftarrow b_{k+1} - [\mu_{k+1,k}] b_k$
        if $\|b_k^*\|^2 > t^2 \|b_{k+1}(k)\|^2$ **then**
            zamenjaj b_{k+1} in b_k
            $k \leftarrow \max(k-1, 1)$
        else
            for $j \leftarrow k-1$ **to** 1 **do**
                $b_{k+1} \leftarrow b_{k+1} - [\mu_{k+1,j}] b_k$
                $k \leftarrow k+1$
            end for
        end if
    end while
    return $(b_1, \ldots, b_n)$.
end procedure

Vsote in rekurzivne enačbe

Def: Zaporedje $a \in K^{\mathbb{N}}$ je **hipergeometrično**, če obstajata $n_0 \in \mathbb{N}$ in $r \in K(n)$ tako da $\forall n \geq n_0 \colon a_n \neq 0 \wedge \frac{a_{n+1}}{a_n} = r(n)$. Množico vseh hipergeometričnih zaporedij označimo s $\mathcal{H}(K)$.

Izr: $\forall r \in K(k)\ \exists a, b, c \in K[k] \colon$
$r(k) = \frac{a(k)}{b(k)} \frac{c(k+1)}{c(k)} \wedge (\forall i \in \mathbb{N}_0 \colon a(k) \perp b(k+i)) \wedge a(k) \perp c(k) \wedge b(k) \perp c(k+1)$

Za dano hipergeometrično zaporedje $t \in K^{\mathbb{N}}$ želimo njegovo vsoto $\sum_{k=k_0}^{n} t_k$ izraziti v zaključeni obliki.

Algoritem 7 Gosperjev algoritem.
Vhod: Hipergeometrično zaporedje t_k, tako da $\frac{t_{k+1}}{t_k} = r(k) \in K(k)$.
Izhod: Hipergeometrično zaporedje s_k, tako da $s_{k+1} - s_k = t_k$ s.p., če obstaja.

procedure GA(t)
    izračunaj $r(k) \leftarrow \frac{t_{k+1}}{t_k}$
    najdi $a, b, c \in K[k] \colon r(k) = \frac{a(k)}{b(k)} \frac{c(k+1)}{c(k)}$, tako da $a(k) \perp b(k+i), i = 0, 1, 2, \ldots$
    najdi $x(k) \in K[k]$, ki reši $a(k)x(k+1) - b(k-1)x(k) = c(k)$. ▷ Lahko uporabimo poli.
    if $x(k)$ ne obstaja **then**
        return rešitev ne obstaja
    else
        return $s_k = \frac{b(k-1)x(k)}{c(k)} t_k$
    end if
end procedure

Z Gosperjevim algoritmom lahko zapišemo vsoto hipergeometričnega zaporedja v zaključeni obliki $\sum_{k=k_0}^{n} t_k = s_{n+1} - s_{k_0}$. Gosperjev algoritem nam odpre nov problem, ki je reševanje linearne rekurzivne enačbe s polinomskimi koeficienti. Tega se bomo lotili z algoritmom Poli (alg. 8).
Hint: Včasih s_{n+1} (ali kateri drugi člen) ni definiran, takrat uporabimo $\sum_{k=k_0}^{n} t_k = s_n - s_{k_0} + t_n$. V splošnem velja $\sum_{k=k_0}^{k_1} t_k = s_{k_1} - s_{k_0} + t_{k_1}$.

Def: Operator $E \colon K^{\mathbb{N}} \to K^{\mathbb{N}}$ se imenuje **operator pomika** in premakne člene zaporedja za eno mesto $(Ea)_n = a_{n+1}$.
Def: Operator $\Delta = E - 1$ je **diferenčni operator**, $(\Delta a)_n = a_{n+1} - a_n$.
Def: Množica $L[n; E] = \left\{ \sum_{i=0}^{r} p_i E^i \mid r \in \mathbb{N}, p_i \in K[n] \right\}$ je kolobar linearnih rekurzivnih operatorjev (LRO) s polinomskimi koeficienti.

Algoritem 8 Algoritem poli.
Vhod: LRO $L = \sum_{j=0}^{r} q_j(n)\Delta^j \in K[n;\Delta]$ in polinom $f \in K[n]$.
Izhod: Baza B afinega prostora $\{x \in K[n] \mid Lx = f\}$.

procedure POLI(t)
  $b \leftarrow \max_{0 \le j \le r} (\operatorname{st} q_j - j)$
  $p(z) \leftarrow \sum_{0 \le j \le r} \operatorname{vk}(q_j) z^{\underline{j}}$
  $\alpha \leftarrow \max(\{k \in \mathbb{N} \mid p(k) = 0\} \cup \{-1\})$
  $d \leftarrow \max(\operatorname{st} f - b, \alpha)$
  if $d < 0$ **then**
    if $f = 0$ **then**
      return $\{0\}$
    else
      return $\{\}$
    end if
  else
    return $B = \{x \in K[n] \mid \operatorname{st} x \le d, Lx = f\}$ ▷ z metodo nedoločenih koeficientov
  end if ▷ za c_i in nastavkom $x(n) = \sum_{i=0}^{d} c_i n^i$
end procedure

Def: $F\colon \mathbb{N} \times \mathbb{N} \to K$ je dvorazsežno hipergeomertijsko zaporedje, če obstajajo $p_1, p_2, q_1, q_2 \in K[n,k] \setminus \{0\}$, tako da $p_1(n,k)F(n+1,k) = q_1(n,k)F(n,k)$ in $p_2(n,k)F(n,k+1) = q_2(n,k)F(n,k)$.
Def: $F\colon \mathbb{N} \times \mathbb{N} \to K$ je *pravo* dvorazsežno hipergeometrijsko zaporedje, če je oblike $F(n,k) = P(n,k)y^n z^k \prod_{i=0}^{p}(a_i n + b_i k + \alpha_i)! / \prod_{j=0}^{q}(c_j n + d_j k + \beta_j)!$ kjer $P \in \mathbb{C}[n,k]$, $y,z \in \mathbb{C}^*$, $a_i,b_i,c_j,d_j \in \mathbb{Z}$, $p,q \in \mathbb{N}$, $\alpha_i,\beta_j \in \mathbb{C}$, tako da $a_i n + b_i k + \alpha_i$ niso negativna cela števila.

Algoritem 9 Zeilbergerjev algoritem (ni nujno, da se konča, razen za *pravo*).
Vhod: Dvorazsežno hipergeometrijsko zaporedje $F(n,k)$.
Izhod: $L \in K(n)[E_n]$ in $G(n,k)$, tako da $(LF)(n,k) = G(n,k+1) - G(n,k)$.

procedure ZA(F)
  for $d \leftarrow 0$ **to** ∞ **do**
    Uporabi GA nad $K(n)$ na produktu $t_k = F(n+d,k) - \sum_{i=0}^{d-1} r_i(n)F(n+i,k)$, kjer so $r_0, \ldots, r_{d-1} \in K(n)$ nedoločene.
    if GA uspe **then**
      Naj GA vrne s_k in $r_0, \ldots, r_{d-1}$, potem vrni $L = E_n^d - \sum_{i=0}^{d-1} r_i(n)F(n+i,k)$, $G(n,k) = s_k$.
    end if
  end for
end procedure

Na koncu ZA moramo rešiti hipergeometrijsko LRE. Za to imamo algoritem Hiper (alg. 10).
Hint: Pogosto se splača čez celotno enačbo narediti $\sum_{k=0}^{\infty}$, saj tako dobimo hipergeometrijsko LRE za $S(n)$ (vsota F), pri tem, ko (npr. v primeru binomskih simbolov v $G(n,k)$) na eni strani dobimo 0. Velja tudi $\binom{0}{0} = 1$ (koristno pri $S(0)$) in če $m > n$ potem $\binom{n}{m} = 0$ (zato je $G(n,\infty)$ pogosto 0).

Znižanje reda enačbe

Če imamo neko hipergeometrično rešitev $La = 0$, potem lahko znižamo red enačbe, tako da gremo noter z nastavkom $y_n = a_n z_n$. To nam da enačbo za z_n: $L' z_n = 0$. Linearni operator L' zapišemo po potencah Δ, prosti člen se mora pokrajšat in dobimo enačbo za $(\Delta z)_n$, ki je enega reda manjša. Potem to rešimo in dobimo rešitev $(\Delta z)_n = b_n$, od tod pa dobimo $z_n = c_1 \sum_{k=k_0}^{n-1} b_k + c_2$, $c \in K$ in od tod končno $y_n = a_n(c_1 \sum_{k=k_0}^{n-1} b_k + c_2)$.

Algoritem 10 Algoritem Hiper (Petkovškov algoritem).
Vhod: $p, q, r \in K[n], pr \neq 0$.
Izhod: Vse hipergeometrične rešitve LRE $p(n)y_{n+2} + q(n)y_{n+1} + r(n)y_n$.

procedure HIPER(p, q, r)
  for all monične $a(n) \mid r(n)$ in $b(n) \mid p(n-1)$ **do**
    $P(n) \leftarrow a(n+1)p(n)/b(n+1)$
    $Q(n) \leftarrow q(n)$
    $R(n) \leftarrow b(n)r(n)/a(n)$
    $\rho \leftarrow \max\{\operatorname{st} P, \operatorname{st} Q, \operatorname{st} R\}$
    $\alpha \leftarrow [n^\rho]P(n)$
    $\beta \leftarrow [n^\rho]Q(n)$
    $\gamma \leftarrow [n^\rho]R(n)$
    for all rešitve $z \neq 0$ enačbe $\alpha z^2 + \beta z + \gamma = 0$ **do**
      for all poli. rešitve $c(n)$ enačbe $z^2P(n)c(n+2) + zQ(n)c(n+1) + R(n)c(n)$ **do**
        $f(n) \leftarrow z\frac{a(n)}{b(n)}\frac{c(n+1)}{c(n)}$
        med vse rešitve dodaj rešitve LRE $y_{n+1} = f(n)y_n$.
      end for
    end for
  end for
end procedure

Računanje v polinomskih idealih

Oznake: $x := (x_1, \ldots, x_n)$, $\alpha := (\alpha_1, \ldots, \alpha_n)$, $|\alpha| := \alpha_1 + \ldots + \alpha_n$, $x^\alpha = x_1^{\alpha_1} \ldots x_n^{\alpha_n}$.
Def: Za $\alpha, \beta \in \mathbb{N}^n$ definiramo relacijo $\subseteq$ kot $\alpha \subseteq \beta \iff \forall i\ \alpha_i \leq \beta_i$. Velja $\alpha \subseteq \beta \implies x^\alpha | x^\beta$.
Def: Relacija $\leq$ v $\mathbb{N}^n$ je **monomska urejenost**, če linearno ureja elemente, so vsi elementi večji od 0 in je tranzitivna. Primeri: $\leq_{LEX}, \leq_{TLEX}, \leq_{TRLEX}$.
Def: Množica monomov $\mathrm{Mon} = \{x^\alpha \mid \alpha \in \mathbb{N}^n\}$. Z dano monomsko urejenostjo $\leq$ za $\mathbb{N}^n$, na Mon definiramo relacijo $\leq$ s predpisom $x^\alpha \leq x^\beta \iff \alpha \leq \beta$.
Lema (DL): $\forall M \subseteq \mathrm{Mon}\ \exists B \subseteq M\colon |B| < \infty \wedge (\forall m \in M\ \exists b \in B\colon b|m)$.

Def: Naj bo $\leq$ monomska urejenost na $\mathbb{N}^n$, $f \in K[x] \setminus \{0\}$ in pripadajoča $A \subseteq \mathbb{N}^n$ končna, tako da $f = \sum_{\alpha \in A} c_\alpha x^\alpha$ in $\forall \alpha \in A\colon c_\alpha \neq 0$. Potem definiramo naslednje oznake:

(i) skupna stopnja f: $\operatorname{st} f := \max_{\alpha \in A} |\alpha|$
(ii) multistopnja f: $\operatorname{mst} f := \max_{\leq} A$
(iii) vodilni koeficient f: $\operatorname{vk}_{\leq} f := c_{\operatorname{mst}_{\leq} f}$
(iv) vodilni monom f: $\operatorname{vm}_{\leq} f := x^{\operatorname{mst}_{\leq} f}$
(v) vodilni člen f: $\operatorname{vč}_{\leq} f := \operatorname{vk}_{\leq} f \operatorname{vm}_{\leq} f$

Izr: Naj bo $\leq$ monomska urejenost in $f_1, \ldots f_k \in K[x] \setminus \{0\}$, $g \in K[x]$. Potem obstajajo $h_1, \ldots, h_k, r \in K[x]$, tako da $g = \sum h_i f_i + r$, noben člen r_i ni deljiv z nodebim izmed členov $f_1, \ldots, f_k$ in, če $h_i \neq 0$, je $\operatorname{mst}(h_i f_i) \leq \operatorname{mst}(g)$. Za algoritem za deljenje glej algoritem 11.

Def: Naj bo $I \subseteq K[x]$ ideal $G \subseteq I$ končna. Če za vsak $f \in I \setminus \{0\}$ obstaja $g \in G$, tako da $\operatorname{vm}(g) \mid \operatorname{vm}(f)$, potem je G **Gröbnerjeva baza** za I.
Trd: Naj bo $\{g_1, \ldots, g_m\}$ Gröbnerjeva baza za I. Potem je $f \in I \iff f \mod (g_1, \ldots, g_m) = 0$ in $I = \langle g_1, \ldots, g_m \rangle$.
Trd: Vsak ideal ima Gröbnerjevo bazo.
Izr: Vsak polinomski ideal je končno generiran.
Def: Za dva neničelna polinoma $f, g \in K[x]$ in monomsko urejenost $\leq$ naj bo S-polinom polinomov f in g polinom $S(f,g) = \frac{m}{\operatorname{vč}(f)} f - \frac{m}{\operatorname{vč}(g)} g$, kjer je $m = \operatorname{lcm}(\operatorname{vm} f, \operatorname{vm} g)$.
Izrek: I ideal v $K[x]$ z bazo $G = \{g_1, \ldots, g_k\}$. Potem je G Gröbnerjeva $\iff \forall g_i, g_j \in G$ velja $S(g_i, g_j) \mod G = 0$. To je temeljni izrek za algoritem 12, ki poišče Gröbnerjevo bazo.

Def: Gröbnerjeva baza je reducirana, če za vse $g \in G$ velja, da za vsak $h \in G \setminus \{g\}$ noben člen h ni deljiv z $\operatorname{vč}(g)$ in $\operatorname{vk}(g) = 1$. Vsak ideal ima enolično reducirano Gröbnerjevo bazo.

Avtorji: Jure Slak, Maks Kolman, Žiga Lukšič

Algoritem 11 Algoritem za deljenje v $K[x_1, \ldots, x_n]$.
Vhod: $\leq$ monomska urejenost, $f_1, \ldots, f_k, g$ kot zgoraj.
Izhod: $h_1, \ldots, h_k, r$ kot zgoraj.

```
1:  procedure DELI(g, f_1, ..., f_k)
2:      for i ← 1 to k do
3:          h_i ← 0
4:      end for
5:      r ← 0
6:      p ← g
7:      while p ≠ 0 do
8:          i ← 1
9:          uspeh ← 0
10:         while i ≤ k  and  uspeh ≠ 1 do
11:             if več f_i | več p then
12:                 p ← p − g več(p)/ več(f_i)
13:                 h_i ← h_i + več(p)/ več(f_i)
14:                 uspeh ← 1
15:             else
16:                 i ← i + 1
17:             end if
18:         end while
19:         if uspeh = 0 then
20:             p ← p − več(p)
21:             r ← r + več(p)
22:         end if
23:     end while
24: end procedure
```

Algoritem 12 Buchbergerjev algoritem iskanje Gröbnerjeve baze.
Vhod: $\leq$ monomska urejenost, $f_1, \ldots, f_k \in K[x]$.
Izhod: Gröbnerjeva baza ideala $\langle f_1, \ldots, f_k \rangle$ glede na $\leq$.

```
1:  procedure GROEBNER(f_1, ..., f_k)
2:      G' ← {f_1, ..., f_k}
3:      repeat
4:          G ← G'
5:          for all pairs (f, g) ∈ G^2 do
6:              r ← S(f, g)  mod G
7:              if r ≠ 0 then
8:                  G' ← G' ∪ {r}
9:              end if
10:         end for
11:     until G' = G
12: end procedure
```