3 skyrius
Rekursyviosios funkcijos

Šiame skyriuje nagrinėsime vieną *algoritmiškai apskaičiuojamųjų funkcijų* formalizmą – rekursyviąsias funkcijas. Visų šiame skyriuje nagrinėjamųjų funkcijų apibrėžimų bei reikšmių aibėmis yra viena ir ta pati *natūraliųjų skaičių aibė* $N = \{0, 1, 2, ...\}$. Rekursyviųjų funkcijų aibė sutampa su Turing mašinomis apskaičiuojamųjų funkcijų aibe. D.Hilbert suformulavo reikalavimus, kuriuos turi tenkinti algoritmiškai apskaičiuojamosios funkcijos. Remiantis jo darbais 1931 metais K.Gödel pirmasis aprašė rekursyviųjų funkcijų klasę. Vėliau, 1936 metais A.Church, panaudodamas kitas idėjas, aprašė tą pačią rekursyviųjų funkcijų klasę.

3.1 Intuityvioji algoritmo samprata

XX amžiaus pradžioje atsirado poreikis tikslios *efektyvios procedūros (algoritmo)* sąvokos apibrėžimui. Pradėta manyti, kad kai kurios problemos nėra išsprendžiamos. Bet kaip tai įrodyti? Kaip įrodyti, kad tam tikrai problemai spręsti nėra algoritmo? Tam nepakanka plačiai vartojamos intuityvios algoritmo sampratos:

griežtų komandų (instrukcijų), pagal kuriuos atliekamos operacijos, seka, leidžianti spręsti matematikos ar logikos uždavinius.

Reikia turėti matematiškai tikslią algoritmo sąvoką. Ji turėtų apibendrinti intuityviai suprantamų algoritmų savybes.

Pateiksime plačiai žinomą Euklido algoritmo pavyzdį. Duoti du natūralieji skaičiai $a_1 \geq a_2 > 0$. Rasti bendrą didžiausią daliklį. Algoritmas toks:

Dalome a_1 iš a_2. Jei liekana $a_3 = 0$, tai a_2 yra bendras didžiausias daliklis. Jei $a_3 \neq 0$, tai atliekame kitą veiksmą.

Dalome a_2 iš a_3. Jei liekana $a_4 = 0$, tai a_3 ir yra bendras didžiausias daliklis. Jei $a_4 \neq 0$, tai atliekame kitą veiksmą.

Dalome a_3 iš a_4 ir t.t

Kadangi $a_1 \geq a_2 > a_3 > ...$, tai po baigtinio žingsnių skaičiaus rasime bendrą didžiausią a_1 ir a_2 daliklį.

Pagrindinės savybės, kurias tenkina žinomų algoritmų pavyzdžiai yra tokios:

1. *Diskretumas.* Veiksmai išdėstyti tam tikra seka. Vieną jų atlikę, pereiname prie kito. Veiksmai dar vadinami algoritmo žingsniais.

2. *Determinuotumas.* Atlikę veiksmą žinome (nurodyta) ką toliau daryti.

3. *Žingsnių elementarumas.* Algoritmo veiksmų seką galima suskaidyti į labai paprastus, elementarius, nesunkiai aprašomus ir nesunkiai įvykdomus žingsnius.

4. *Masiškumas.* Algoritmai taikomi tam tikros aibės atžvilgiu. Pavyzdžiui, aprašytasis Euklido algoritmas taikomas *bet kuriems* natūraliesiems skaičiams $a_1 \geq a_2 > 0$.

Iš kur gi kilo žodis algoritmas? Tai sulotynintas arabų (kai kurie šaltiniai nurodo, persų) matematiko Al Chorezmi (gimė apie 783 metus, mirė apie 850 metus) vardas. Jis išgarsėjo savo knyga, kurioje aprašė veiksmus su iš Indijos perimtais skaičiais. Naujoji pozicinė skaičiavimo sistema greitai paplito pasaulyje, o jo knyga tapo daugelio žmonių parankine knyga. Knygoje buvo daug taisyklių rinkinių, kurių pagalba po baigtinio žingsnių skaičiaus gauname rezultatą.

Tikslinant algoritmo sąvoką buvo einama dviem keliais:

1. Idealizuotos (matematinės) skaičiavimo mašinos kūrimas,

2. Algoritmiškai apskaičiuojamų funkcijų aibės aprašymas.

Po daugelio metų, pagaliau 1934-1936 metais ir viena ir kita kryptimi dirbančių mokslininkų dėka gauta daug skirtingų savo formalizmu bei idėjomis algoritmo sąvokos patikslinimų. Pirmosios krypties žinomiausiais darbais tapo A.Turing ir E.Post aprašytosios mašinos. A.Turing laikomas *informatikos mokslo tėvu.* Sukurtąją mašiną pasiūlė vadinti *elektroniniu kompiuteriu.* Antrojo pasaulinio karo metu jis panaudojo tokią mašiną, iššifruodamas vokiečių naudotą povandeninių laivų kodą *Enigma.* 1954 metais baigė savo gyvenimą nusinuodydamas kalio cianidu. Paskutiniuosius savo gyvenimo metus A.Turing dirbo Mančesterio universitete. Intuityviai *algoritmiškai apskaičiuojamoji funkcija* suprantama taip: žinant funkcijos $y = f(x)$ argumento reikšmę *moku apskaičiuoti* funkcijos reikšmę. Buvo sukurta daug metodų *algoritmiškai apskaičiuojamųjų funkcijų* klasei nusakyti. Žinomiausios yra K.Gödel, A.Church bei S.Kleene aprašytosios funkcijų klasės. A.Church jas pavadino rekursyviosiomis funkcijomis.

A.Church tezė. *Algoritmiškai apskaičiuojamųjų funkcijų aibė sutampa su rekursyviųjų* *funkcijų aibe.*

Ji buvo paskelbta 1936 metais. Teze vadinama todėl, kad tai tvirtinimas, kuriuo, A.Church nuomone, tikėti reikėtų, bet įrodyti negalima. Negalima įrodyti dviejų aibių lygybės, nes, iš vienos pusės, tai matematiškai tiksli rekursyviųjų funkcijų klasė, o iš kitos − intuityvi, netiksli, skirtingų žmonių suprantama skirtingai, algoritmiškai apskaičiuojamųjų funkcijų klasė.

Kodėl tikima A.Church teze? Pagrindiniai argumentai yra du:

1. Visų pasiūlytų, skirtingomis idėjomis aprašytų algoritmiškai apskaičiuojamųjų funkcijų klasės sutampa ir ne tik tarpusavyje, bet ir su idealizuotų skaičiavimo mašinų apskaičiuojamosiomis funkcijų klasėmis,

2. Nėra žinomas joks intuityviai apskaičiuojamosios funkcijos pavyzdys, kuris nebūtų rekursyvioji funkcija.

3.2 Primityviai rekursinės funkcijos

Aprašysime formaliąją sistemą. Funkcijos, kurias galima gauti toje sistemoje, ir vadinsime rekusyviosiomis. Formalioji sistema susideda iš bazinių funkcijų ir operatorių, kurie taikomi turimoms funkcijoms, o rezultatas − naujosios funkcijos. Visų pirma aprašysime vieną poaibį rekursyviųjų funkcijų, taip vadinamąsias primityviai rekursines funkcijas.

Bazinės funkcijos: konstanta 0, paskesniojo nario funkcija $s(x) = x + 1$ ir projekcijų funkcijos $pr_p^i(x_1, ..., x_p) = x_i$ $(p \geq 1; 1 \leq i \leq p)$.

Iš bazinių funkcijų gauname naujas naudojantis šiais dviem operatoriais:

1. **Kompozicijos operatorius.** Tarkime duotos $n + 1$ funkcijos

$$f(x_1, ..., x_n), g_1(x_1^1, ..., x_{m_1}^1), ..., g_n(x_1^n, ..., x_{m_n}^n) \tag{3.1}$$

Sakysime, kad funkcija $f(g_1(x_1^1, ..., x_{m_1}^1), ..., g_n(x_1^n, ..., x_{m_n}^n))$ gauta iš funkcijų (3.1) panaudojus kompozicijos operatorių.

2. **Primityviosios rekursijos operatorius.** Tarkime duotos dvi funkcijos $g(x_1, ..., x_{n-1}), h(x_1, ..., x_{n+1})$. Viena jų yra $n - 1$ argumento, o antroji $n + 1$ argumento. Apibrėžiame naująją n argumentų funkciją $f(x_1, ..., x_n)$ tokia shema:

$$f(x_1, ..., x_{n-1}, 0) = g(x_1, ..., x_{n-1})$$
$$f(x_1, ..., x_{n-1}, y + 1) = h(x_1, ..., x_{n-1}, y, f(x_1, ..., x_{n-1}, y))$$

Kai $n = 1$ funkcija g yra konstanta. Sakysime, kad funkcija f gauta iš funkcijų *g,h*, panaudojus primityviosios rekursijos operatorių. Pabrėždami, kad paskutiniojo argumento reikšmė nelygi nuliui, rašome $y + 1$. Kaip matome, funkcija apibrėžta rekursyviai. Norint apskaičiuoti funkcijos $f(x_1, ..., x_{n-1}, y + 1)$ reikšmę, reikia iš pradžių rasti funkcijos reikšmę, kai paskutiniojo argumento reikšmė vienetu mažesnė. Rekursija (grįžimas) primityvi. Argumento reikšmė sumažinama vienetu.

3.1 apibrėžimas. *Pati mažiausia aibė, kuriai priklauso bazinės funkcijos ir kuri uždara kompozicijos bei primityviosios rekursijos atžvilgiu, vadinama primityviai rekursinių funkcijų aibe (klase).*

Primityviai rekursinių funkcijų aibę žymėsime *PR*. Primityviai rekursinės funkcijos apibrėžtos su bet kuriomis argumentų reikšmėmis. Tokias funkcijas vadinsime *visur apibrėžtomis funkcijomis.*

Aibės A charakteringoji funkcija κ apibrėžiama tokiu būdu:

$$\kappa_A\ (x) = \begin{cases} 1, & jei \quad x \in A \\ 0, & jei \quad x \notin A \end{cases}$$

3.2 apibrėžimas. *Natūraliųjų skaičių poaibis vadinamas primityviai rekursiniu, jei jo charakteringoji funkcija yra primityviai rekursinė.*

Pavyzdžiai:

1. Konstanta 1 priklauso *PR* klasei, nes ją $s(0)$ galima gauti iš bazinių funkcijų $s(x)$, 0, naudojantis kompozicijos operatoriumi.

2. Bet kuri konstanta n priklauso *PR* klasei, nes $s(s(...s(0))...) \in PR$. Čia taikėme $n - 1$ kartą kompoziciją.

3. $x + n \in PR$, nes $s(s(...s(x))...) = x + n$. Taip pat taikėme $n - 1$ kartą kompoziciją.

4. $s(pr_3^3(x, y, z)) \in PR$. Gauta iš bazinių funkcijų $s(x), pr_3^3(x, y, z)$, pritaikius kompozicijos operatorių.

5. Parodysime, kad $x + y \in PR$. Funkcija $x + y$ gaunama iš pr_1^1 bei $s(pr_3^3(x, y, z))$ naudojantis primityviosios rekursijos operatoriumi.

$x + 0 = pr_1^1(x) = x$
$x + (y + 1) = (x + y) + 1 = s(pr_3^3(x, y, x + y)) = s(x + y) = (x + y) + 1$

Apibrėžiame funkcijas $sg\ x$, $\bar{sg}\ x$, bei $x \dot{-} y$:

$$sg\ x = \begin{cases} 1, & jei \quad x > 0 \\ 0, & jei \quad x = 0 \end{cases}$$

$$\bar{sg}\ x = \begin{cases} 1, & jei \quad x = 0 \\ 0, & jei \quad x > 0 \end{cases}$$

$$x \dot{-} y = \begin{cases} x - y, & jei \quad x > y \\ 0, & jei \quad x \leq y \end{cases}$$

Įrodymą, kad $sg\ x$, $\bar{sg}\ x$ bei $x \dot{-} y$ yra primityviai rekursinės, paliekame pratyboms.

Funkcija $|x - y|$ irgi primityviai rekursinė, nes $|x - y| = (x \dot{-} y) + (y \dot{-} x)$.

3.1 lema *Jei $g(x_1, ..., x_n)$ primityviai rekursinė, tai*

$$f(x_1, ..., x_n) = \sum_{i=0}^{x_n} g(x_1, ..., x_{n-1}, i)$$

irgi primityviai rekursinė.

Įrodymas.
$f(x_1, ..., x_{n-1}, 0) = g(x_1, ..., x_{n-1}, 0)$
$f(x_1, ..., x_{n-1}, y + 1) = f(x_1, ..., x_{n-1}, y) + g(x_1, ..., x_{n-1}, y + 1)$

Todėl $f(x_1, ..., x_n)$ gaunama pritaikius primityviosios rekursijos operatorių funkcijoms $g(x_1, ..., x_{n-1}, 0)$ bei $h(x_1, ..., x_{n+1}) = g(x_1, ..., x_{n-1}, s(x_n)) + x_{n+1}$, kurios yra primityviai rekursinės. Lema įrodyta.

Dalome x iš y. Sveikąją dalį žymime $[x/y]$, o liekaną $rest(x, y)$. Tarkime, kad $[x/0] = x$, bei $rest(x, 0) = x$. Remiantis 3.1 lema nesunku įrodyti, kad $[x/y]$ yra primityviai rekursinė. Tuo tikslu nagrinėjame skaičių seką:

$1 \cdot y \dot{-} x$, $2 \cdot y \dot{-} x$,..., $n \cdot y \dot{-} x$, ..., $x \cdot y \dot{-} x$.

Sveikoji dalis lygi nulių skaičiui sekoje. Todėl

$$[x/y] = \sum_{i=0}^{x} \bar{sg}(iy \dot{-} x) \dot{-} 1.$$

$$rest(x, y) = x \dot{-} (y \cdot [x/y]).$$

Algoritmų teorijoje dažnai sutinkamas funkcijos *apibrėžimas dalimis.*

$$f(x_1,...,x_n) = \begin{cases} f_1(x_1,...,x_n), & jei \quad \alpha_1(x_1,...,x_n) = 0 \\ ... & \\ f_s(x_1,...,x_n), & jei \quad \alpha_s(x_1,...,x_n) = 0 \\ f_{s+1}(x_1,...,x_n), & likusiais \ atvejais \end{cases}$$

Be to, su bet kuriuo reikšmių $(x_1,...,x_n)$ rinkiniu, tik viena iš α_i gali būti lygi nuliui.

Jei funkcijos f_i $(i = 1,...,s+1)$ bei α_i $(i = 1,...,s)$ primityviai rekursinės, tai ir f primityviai rekursinė, nes teisinga lygybė:

$$f(x_1,...,x_n) = f_1(x_1,...,x_n) \cdot \bar{sg}\alpha_1(x_1,...,x_n) + ... + f_s(x_1,...,x_n) \cdot \bar{sg}\alpha_s(x_1,...,x_n) +$$
$$+ f_{s+1}(x_1,...,x_n) \cdot sg(\alpha_1(x_1,...,x_n) \cdot ... \cdot \alpha_s(x_1,...,x_n))$$

Sąlygas α_i galima pakeisti tokiomis (suprantama α_i, β_i primityviai rekursinės)

$$\alpha_i = \beta_i,\ \alpha_i \leq \beta_i,\ \alpha_i < \beta_i,$$

nes jos redukuojamos į lygtis

$$|\alpha_i - \beta_i| = 0,\ \alpha_i \dot{-} \beta_i = 0,\ \bar{sg}(\beta_i \dot{-} \alpha_i) = 0.$$

3.3 Minimizacijos operatorius

Tarkime duota n argumentų funkcija f. Apibrėžiame naują, taip pat n argumentų, funkciją $g(x_1,...,x_n)$, kurios reikšmė lygi mažiausiam y, su kuriuo $f(x_1,...,x_{n-1},y) = x_n$. Įrodyta, kad net jeigu f yra primityviai rekursinė, g gali ir nebūti algoritmiškai apskaičiuojamoji funkcija. Todėl nusakydami naują funkciją g, mes privalome nurodyti ir metodą kaip bus ieškomas mažiausias y:

Jei $f(x_1,...,x_{n-1},0) = x_n$, *tai funkcijos* g *reikšmė lygi 0, jei ne, tai tikriname ar* $f(x_1,...,x_{n-1},1) = x_n$.
Jei $f(x_1,...,x_{n-1},1) = x_n$, *tai funkcijos* g *reikšmė lygi 1, jei ne, tai tikriname ar* $f(x_1,...,x_{n-1},2) = x_n$ *ir t.t.*

Funkcija g gali būti ir dalinė, t.y. su kai kuriomis argumentų reikšmėmis ji gali būti ir neapibrėžta, nes, pavyzdžiui, tokio y, tenkinančio aprašytąją lygybę, gali ir nebūti. Antra vertus, gali ir būti toks m, kad $f(x_1,...,x_{n-1},m) = x_n$, bet, jei su kuriuo nors $i < m$ funkcija $f(x_1,...,x_{n-1},i)$ neapibrėžta, tai ir g bus neapibrėžta.

Sakysime, kad g gauta pritaikius minimizacijos operatorių funkcijai f, ir žymėsime

$$g(x_1,...,x_n) = \mu_y(f(x_1,...,x_{n-1},y) = x_n).$$

Naudojantis minimizacijos operatoriumi gaunama dalinė skirtumo funkcija $x-y=\mu_z(y+z=x)$.

Funkcija $f(x)=\mu_y(y-(x+1)=0)$ neapibrėžta su jokiu $x\in N$, nors kiekvienam x atsiras mažiausias y. Jis lygus $x+1$.

3.3 apibrėžimas. *Pati mažiausia aibė, kuriai priklauso bazinės funkcijos ir kuri uždara kompozicijos, primityviosios rekursijos bei minimizacijos atžvilgiu, vadinama dalinai rekursinių funkcijų aibe (klase).*

$x-y$ yra dalinai rekursinė funkcija, bet ji nėra primityviai rekursinė. Iš 3.1 ir 3.3 apibrėžimų išplaukia, kad kiekviena primityviai rekursinė funkcija yra ir dalinai rekursinė.

3.4 apibrėžimas. *Visur apibrėžta dalinai rekursinė funkcija vadinama bendrarekursine funkcija.*

Dalinai rekursinių funkcijų aibę žymėsime DR, o bendrarekursinių $-$ BR. Iš apibrėžimų išplaukia, kad $PR\subseteq BR\subset DR$. Vėliau parodysime, kad egzistuoja bendrarekursinės funkcijos, kurios nėra primityviai rekursinės. Tokiu būdu $PR\subset DR\subset BR$.

3.4 Porų numeracija

Kadangi bet kurių dviejų skaičiųjų aibių Dekarto sandauga skaičioji, tai aibė $A=\{(x,y) : x,y\in N\}$ taip pat skaičioji. Išrašysime visas poras tam tikra tvarka. Jei $x+y<u+v$, tai (x,y) sekoje bus sutinkama anksčiau negu kad (u,v). Jei $x+y=u+v$ ir $x<u$, tai pora (x,y) taip pat sutinkama anksčiau. Turime tokią porų seką:

$(0,0)$, $(0,1)$, $(1,0)$, $(0,2)$, $(1,1)$, $(2,0)$, $(0,3)$,...

Kiekvienai porai priskirsime po numerį. Numeraciją pradėsime nuo nuliaus. Jei pora yra i-toje vietoje, tai jos numeris bus $i-1$. Poros (x,y) numerį žymėsime $\alpha_2(x,y)$. Tokiu būdu $\alpha_2(0,0)=0$, $\alpha_2(0,1)=1$, $\alpha_2(1,0)=2$,... Tokią porų numeraciją pasiūlė G.Cantor.

3.2 lema. *Poros (x,y) numeris apskaičiuojamas funkcija*

$$\alpha_2(x,y)=\frac{(x+y)^2+3x+4}{2}.$$

Įrodymas. Pora *(x,y)* yra atkarpoje

$(0,x+y)$, $(1,x+y-1)$,..., (x,y),..., $(x+y,0)$

Prieš atkarpą yra viena tokia pora *(u,v)*, kad $u + v = 0$, dvi poros *(u,v)*, kuriose $u + v = 1$ ir t.t. Iš viso $1 + 2 + ... + (x + y)$ porų t.y.

$$\frac{(x + y)(x + y + 1)}{2}$$

porų. Pora *(x,y)* yra nagrinėjamosios atkarpos $x + 1$-oje pozicijoje. Kadangi numeracija prasideda nuo nulio, tai

$$\alpha_2\ (x, y) = \frac{(x + y)(x + y + 1)}{2} + x = \frac{(x + y)^2 + 3x + 4}{2}$$

Lema įrodyta.

Poros*(x,y)* kairiuoju nariu vadinsime x, o dešiniuoju $-$ y. Kairiojo nario funkciją žymėsime π_2^1, o dešiniojo π_2^2. Jos abi yra vieno argumento funkcijos. Jei poros *(x,y)* numeris n, tai $\pi_2^1(n) = x$, o $\pi_2^2(n) = y$. Jos tenkina tokias savybes:

$\alpha_2(\pi_2^1(n), \pi_2^2(n)) = n,$
$\pi_2^1(\alpha_2(x, y)) = x,$
$\pi_2^2(\alpha_2(x, y)) = y.$

Žemiau mes nesinaudosime šia konkrečia Cantor numeracija. Svarbu mums tik faktas, kad galima tokia porų numeracija, kai $\alpha_2(x, y), \pi_2^1(n), \pi_2^2(n)$ yra primityviai rekursinės funkcijos.

$\pi_2^1(n)$ lygi funkcijai

$$\pi_2^1(n) = x = n \dot{-} \frac{1}{2}[\frac{[\sqrt{8n+1}] + 1}{2}][\frac{[\sqrt{8n+1}] \dot{-} 1}{2}]$$

Naudojantis porų numeracijos funkcija aprašomos trejetų, ketvertų ir t.t. numeracijos.

$$\alpha_3(x_1, x_2, x_3) = \alpha_2(x_1, \alpha_2(x_2, x_3))$$

Pagal šią numeraciją pirmaisiais penkiais sekos nariais yra trejetai:

$$(0, 0, 0), (0, 0, 1), (1, 0, 0), (0, 1, 0), (1, 0, 1), ...$$

Bet kurio n komponentų vektoriaus numeris apibrėžiamas rekursija

$$\alpha_n(x_1, ..., x_n) = \alpha_2(x_1, \alpha_{n-1}(x_2, ..., x_n))$$

Jei $\alpha_n(x_1, ..., x_n) = x$, tai $\pi_n^i(x) = x_i$. Gauname, kad

$x_1 = \pi_2^1(x)$,
$x_2 = \pi_2^1(\pi_2^2(x))$,
$x_3 = \pi_2^1(\pi_2^2(\pi_2^2(x)))$,
...
$x_{n-1} = \pi_2^1(\pi_2^2(\pi_2^2(...(\pi_2^2(x)))...)$ π_2^2 įeina $n-2$ kartus,
$x_n = \pi_2^2(\pi_2^2(\pi_2^2(...(\pi_2^2(x)))...)$ π_2^2 įeina $n-1$ kartą.

Kadangi α_n, π_n^i funkcijos (jos vadinamos Cantor funkcijomis) gaunamos pritaikius kompozicijos operatorių primityviai rekursinėms funkcijoms, tai ir jos pačios yra primityviai rekursinės.

3.5 Baigtinumo problema

Nagrinėjame vienajuostes determinuotasias Turing mašinas. Be to, laikome, kad jos tenkina sąlygas (tokias mašinas vadinsime standartinėmis):

a) kai mašina po baigtinio žingsnių skaičiaus baigia darbą, t.y. patenka į galutinę būseną, jos skaitančioji galvutė turi būti ties pirmaja (iš kairės) netuščia ląstele,

b) abėcėlėje, be tuščios ląstelės simbolio b, yra dar du (0,1). Be to, pradiniai duomenys bei rezultatai yra dvejetainiai skaičiai.

Būsenas, kaip ir įprasta, žymėsime raidėmis q su indeksais, perėjimų funkciją raide δ, o perėjimų komandas $\delta(q_i, x) = (q_j, x', Y)$ (čia Y žymime vieną iš raidžių *K,D,N*).

Yra žinoma, kad kokia bebūtų Turing mašina, galima rasti jai ekvivalenčią, t.y. apskaičiuojančią tą pačią funkciją, standartinę. Tokiu būdu, Turing mašinos abėcėlė yra tokia:

$A = \{0, 1, b, q, 2, ..., 9, \delta, =, (,), K, D, N\} \cup \{, \}$.

3.3 lema *Standartinių Turing mašinų aibė skaičioji.*

Įrodymas. Sutinkamai su 1.5 teorema, visų galimų žodžių aibė A^* skaičioji. Tie žodžiai, kurie yra kurios nors standartinės Turing mašinos perėjimų funkcija, sudaro begalinį A^* poaibį, kuris yra skaitusis, nes bet kuris skaičiosios aibės poaibis yra baigtinis arba skaitusis. Lema įrodyta.

Tarkime $T_0, T_1, T_2, ...$ – pilnas sąrašas standartinių Turing mašinų, o $\varphi_0, \varphi_1, \varphi_2, ...$ – vieno argumento dalinai rekursinės funkcijos, kurias apskaičiuoja atitinkamos Turing mašinos, t.y. φ_i žymi funkciją, kurią apskaičiuoja mašina T_i.

Baigtinumo problema:
Ar egzistuoja algoritmas, kuriuo naudojantis pagal bet kurią natūraliųjų skaičių porą

(m,n) galima pasakyti, ar Turing mašina T_m su pradiniais duomenimis n (t.y. juostoje pradiniu laiko momentu yra natūralųjį n atitinkantis dvejetainis skaičius) baigia darbą (t.y. po baigtinio žingsnių skaičiaus pereina į galutinę būseną) ar ne?

Jei dalinė funkcija *f(x)* apibrėžta su $x = k$, tai žymėsime $f(k) < \infty$, jei ne, tai $f(k) = \infty$.

Baigtinumo problemą galima apibrėžti ir tokiu būdu: *ar egzistuoja algoritmas, kuriuo galima nustatyti ar $\varphi_m(n) < \infty$, ar $\varphi_m = \infty$ (m, n – bet kurie natūralieji skaičiai)?*

Aibė vadinama *rekursine*, jei jos charakteringaja funkcija yra kuri nors visur apibrėžta rekursyvioji funkcija. Kai kalbama ne apie skaitines aibes, o aibes susidedančias iš kitokių elementų (formulių, funkcijų, Turing mašinų,...), tai dažniausiai vietoje *(ne)rekursinė aibė* sakoma *(ne)išsprendžiama aibė (klasė, problema).*

3.1 teorema. *Baigtinumo problema neišsprendžiama.*

Įrodymas. Parodysime, kad tarp visų galimų algoritmų nėra tokio, kuris išsprendžia baigtinumo problemą. Nagrinėjame funkciją

$$g(\alpha_2(x,y)) = \begin{cases} 1, & jei \quad \varphi_x(y) < \infty \\ 0, & jei \quad \varphi_x(y) = \infty \end{cases}$$

Tarkime, kad algoritmas, apie kurį kalbama baigtinumo problemoje, egzistuoja. Taigi, atsiras tokia standartinė Turing mašina, kad su pradiniais duomenimis $\alpha_2(x, y)$, po baigtinio žingsnių skaičiaus mašina pereis į galutinę būseną ir juostoje bus tik viena natuščia ląstelė. Joje bus 1 arba 0 ir ties ja bus skaitančioji galvutė. Tuomet funkcija $g(\alpha_2(x, y))$ bendrarekursinė ir atsiras Turing mašina apskaičiuojanti funkciją $\psi(x)$:

$$\psi(x) = \begin{cases} 1, & jei \quad g(\alpha_2(x,x)) = 0 \\ \infty, & jei \quad g(\alpha_2(x,x)) = 1 \end{cases}$$

Ją galima gauti tokiu būdu. Kiekvieną galutinę būseną q_i išbraukiame iš galutinių būsenų sąrašo, o perėjimų funkciją papildome:
$\delta(q_i, 1) = (q_i, 1, D)$
$\delta(q_i, b) = (q_i, b, D)$
$\delta(q_i, 0) = (q_{k_i}, 1, N)$

q_{k_i} – kurios nors naujos būsenos. Jas priskiriame galutinių būsenų aibei. Tarkime l yra Turing mašinos, apskaičiuojančios ψ, kuris nors numeris. Tuomet l bus ir ψ numeris, t.y. $\psi = \varphi_l$. Aiškinames, ar $\psi(l) < \infty$ ir iš prielaidos, kad egzistuoja algoritmas, apie kurį kalbama baigtinumo problemoje, gauname prieštaravimą.

1) Tarkime, kad $\psi(l) < \infty$. Tuomet $g(\alpha_2(l, l)) = 0$ ir $\varphi_l(l) = \infty$ t.y. $\psi(l) = \infty$.
2) Tarkime, kad $\psi(l) = \infty$. Tuomet $g(\alpha_2(l, l)) = 1$ ir $\varphi_l(l) < \infty$ t.y. $\psi(l) < \infty$.

Teorema įrodyta.

1953 metais H.G.Rice įrodė bendresnę teoremą:

Tarkime X dalinai rekursinių vieno argumento funkcijų aibė. Jei X netuščia ir nesutampa su visų dalinai rekursinių vieno argumento funkcijų aibe, tai

$$A = \{x : x \in N \ \ ir \ \ \varphi_x \in X\}$$

nerekursinė.

Remiantis Rice teorema galima gauti daug nerekursinių aibių. Pateiksime porą jų:

a) X susideda iš visų vieno argumento tapačiai lygių nuliui primityviai rekursinių funkcijų,

b) X susideda iš visų bendrarekursinių vieno argumento funkcijų.

Neišsprendžiama ir tokia problema: *ar bet kurios dvi Turing mašinos apskaičiuoja vieną ir tą pačią dalinai rekursinę funkciją.*

3.6 Rekursyviai skaičiosios aibės

Pateikiame tris skirtingus rekursyviai skaičiosios aibės apibrėžimus.

3.5 apibrėžimas. *Sakysime, kad aibė rekursyviai skaičioji, jei ji sutampa su kurios nors dalinai rekursinės funkcijos apibrėžimo sritimi.*

3.6 apibrėžimas. *Netuščia aibė rekursyviai skaičioji, jei ji sutampa su kurios nors primityviai rekursinės funkcijos reikšmių aibe.*

3.7 apibrėžimas. *Aibė A rekursyviai skaičioji, jei egzistuoja tokia primityviai rekursinė funkcija f(a,x), kad lygtis $f(a, x) = 0$ turi sprendinį x tada ir tiktai tada, kai $a \in A$.*

Visi trys apibrėžimai, netuščios aibės atveju, ekvivalentūs. Įrodysime tai tik keliems atvejams.

1. Tarkime aibė A rekursyviai skaičioji pagal 3.6 apibrėžimą, t.y. ji netuščia ir yra tokia primityviai rekursinė funkcija *h(x)*, kad $A = \{h(0), h(1), h(2), ...\}$. Parodysime, kad egzistuoja tokia dalinai rekursinė funkcija, kurios apibrėžimo sritis sutampa su A (3.5 apibrėžimas).
Tokia funkcija bus

$$f(x) = \mu_z(h(z) = x)$$

$f(x)$ dalinai rekursinė, nes gauta pritaikius minimizacijos operatorių primityviai rekursinei funkcijai. Be to, jei $x \in A$, t.y. $x = h(i)$, tai atsiras toks $j \leq i$, kad $h(j) = x$, o tai reiškia, kad *f(x)* apibrėžta. Jei $x \notin A$, tai su bet kuriuo i $h(i) \neq x$ ir $f(x)$ neapibrėžta.

2. Tarkime aibė A rekursyviai skaičioji pagal 3.6 apibrėžimą. A sutampa su primityviai rekursinės funkcijos *h(x)* reikšmių aibe. $A = \{h(0), h(1), h(2), ...\}$. Tuomet $|h(x) - a| = 0$ primityviai rekursinė (žiūrėk skyrelį *primityviai rekursinės funkcijos*) ir ji turi sprendinį tada ir tiktai tada, kai $a \in A$. Taigi, A rekursyviai skaičioji pagal 3.7 apibrėžimą. $f(a, x) = |h(x) - a|$.

3. Tarkime egzistuoja tokia primityviai rekursinė funkcija *f(a,x)*, kad lygtis $f(a, x) = 0$ turi sprendinį tada ir tiktai tada, kai $a \in A$ t.y. A rekusyviai skaičioji pagal 3.7 apibrėžimą. A netuščia ir, tarkime *d*, kuris nors jos elementas. Nagrinėjame funkciją

$$h(t) = \pi_2^1(t)\bar{sg}f(\pi_2^1(t), \pi_2^2(t)) + d \cdot sgf(\pi_2^1(t), \pi_2^2(t)).$$

$h(t)$ primityviai rekursinė, nes gauta pritaikius kompozicijos operatorių primityviai rekursinėms funkcijoms. Tarkime $a \in A$, x_0 yra lygties $f(a, x_0) = 0$ sprendinys ir $t_0 = \alpha_2(a, x_0)$. Tuomet $\bar{sg}f(\pi_2^1(t_0), \pi_2^2(t_0)) = \bar{sg}f(a, x_0) = 1$, $sgf(\pi_2^1(t_0), \pi_2^2(t_0)) = 0$ ir $h(t_0) = \pi_2^1(t_0) = a$ t.y. $h(t_0) \in A$.
Tarkime $a \notin A$. Tuomet su bet kuriuo x_0 $f(a, x_0) \neq 0$. Koks bebūtų $t_0 = \alpha_2(a, x_0)$ $\bar{sg}f(\pi_2^1(t_0), \pi_2^2(t_0)) = 0$. Tuo tarpu $sgf(\pi_2^1(t_0), \pi_2^2(t_0)) = 1$ su bet kuriuo $t_0 = \alpha_2(a, x_0)$ ir $h(t_0) = d$, t.y. $h(t_0) \in A$. Taigi A rekursyviai skaičioji pagal 3.6 apibrėžimą.

Norėdami atkreipti dėmesį į rekursinių ir rekursyviai skaičiųjų aibių skirtumą, pateiksime dar tokį apibrėžimą (palyginkite jį su 3.2 apibrėžimu).

Tarkime $\kappa_A(x)$ dalinai rekursinė funkcija tenkinanti sąlygą

$$\kappa_A(x) = \begin{cases} 1, & jei \quad x \in A \\ \infty, & jei \quad x \notin A \end{cases}$$

Tuomet A rekursyviai skaičioji.

Kuo skiriasi skaičiosios nuo rekursyviai skaičiųjų aibių? Tai paaiškės vėliau. Kol kas tik pastebėsime, kad jei A yra skaičioji ir abipusiai vienareikšmę atitiktį tarp N ir A galime nusakyti kuria nors primityviai rekursine funkcija *h(x)* $(A = \{h(0), h(1), ...\})$, tai A taip pat ir rekursyviai skaičioji (pagal 3.6 apibrėžimą). Kiekvienas natūraliųjų skaičių aibės poaibis yra baigtinis arba skaitusis. Vėliau matysime, kad egzistuoja begaliniai natūraliųjų skaičių poaibiai, kurie nėra rekursyviai skaitūs.

Rekursyviai skaičiųjų aibių pavyzdžiai:

1. Tuščia aibė rekursyviai skaičioji, nes ji sutampa su dalinai rekursinės funkcijos $\mu_z(z + (x + 1) = 0)$ (ji su jokia reikšme neapibrėžta) apibrėžimo sritimi (naudojames 3.5 apibrėžimu).

2. $N_- = \{1, 2, 3, ...\}$ rekursyviai skaičioji, nes sutampa su primityviai rekursinės funkcijos *s(x)* reikšmių aibe.

3. Baigtinio skaičiaus rekursyviai skaičiųjų aibių sąjunga bei sankirta yra rekursyviai skaičiosios aibės.
Tarkime $A_1, A_2, ..., A_n$ rekursyviai skaičiosios aibės. Egzistuoja (3.7 apibrėžimas) tokios primityviai rekursinės funkcijos $f_i(a, x)$, kad $f_i(a, x) = 0$ turi sprendinį tada ir tiktai tada, kai $a \in A_i$.
a) Sankirtos atveju konstruojame tokią primityviai rekursinę funkciją

$$f(a, x) = f_1(a, \pi_n^1(x)) + ... + f_n(a, \pi_n^n(x))$$

Su reikšmėmis $x_1^0, ..., x_n^0$ galioja lygybės $f(a, x_i^0) = 0$ $(i = 1, ..., n)$ tada ir tiktai tada, kai atsiras $a \in A_1 \cap A_2 \cap ... \cap A_n$. Tarkime $x^0 = \alpha_n(x_1^0, ..., x_n^0)$. Tuomet $f(a, x^0) = 0$.
b) Sąjungos atveju

$$f(a, x) = f_1(a, \pi_n^1(x)) \cdot ... \cdot f_n(a, \pi_n^n(x))$$

Kai kurios rekursyviai skaičiųjų aibių savybės.

1. Kiekviena rekursinė aibė yra rekursyviai skaičioji.
Tarkime $\kappa_A(x)$ bendrarekursinė charakteringoji aibės A funkcija. Tuomet A sutampa su dalinai rekursinės funkcijos $f(x) = \kappa_A(x) - 1$ apibrėžimo sritimi.

2. Baigtinės aibės yra rekursinės, o tuo pačiu ir rekursyviai skaičiosios.
Tarkime $A = \{a_1, ..., a_m\}$. Tuomet primityviai rekursinė funkcija
$\kappa_A(x) = \bar{sg}(|x - a_1| \cdot |x - a_2| \cdot ... \cdot |x - a_m|)$ yra aibės A charakteringoji.

3. Jei kuri nors aibė A ir jos papildinys $\bar{A}$ (iki natūraliųjų skaičių aibės) rekursyviai skaičiosios aibės, tai kaip A, taip ir $\bar{A}$ yra rekursinės.
Tarkime, A sutampa su primityviai rekursinės funkcijos $f(x)$ reikšmių aibe, o $\bar{A}$ su $g(x)$ reikšmių aibe (remiames 3.6 apibrėžimu). Funkcija

$$h(x) = \mu_z(|f(z) - x| \cdot |g(z) - x| = 0)$$

apibrėžta su bet kuriuo natūraliuoju x, nes $A \cup \bar{A} = N$ ir todėl ji bendrarekursinė. A bei $\bar{A}$ charakteringomis funkcijomis bus šios bendrarekursinės funkcijos:

$\kappa_A(x) = \bar{sg}|f(h(x)) - x|$,
$\kappa_{\bar{A}}(x) = \bar{sg}|g(h(x)) - x|$.

Iš trečiosios savybės išplaukia:

3.2 teorema. *Jei kuri nors rekursyviai skaičioji aibė nėra rekursinė, tai jos papildinys nėra nei rekursinė, nei rekursyviai skaičioji aibė.*

Ši teorema labai svarbi logikai. Formulėms priskiriami numeriai ir aprašytosios sąvokos bei rezultatai taikomi ir formulių aibėms. Vėliau matysime, kad rekursyviai skaičiosiomis bet nerekursinėmis aibėmis yra *tapačiai teisingų* bei *tapačiai klaidingų predikatų logikos formulių aibės.* Iš 3.2 teoremos išplaukia, kad įvykdomų predikatų logikos formulių aibė nėra nei rekursinė, nei rekursyviai skaičioji. Tas pats galioja ir formulių, kurios nėra tapačiai teisingos, aibei.

3.7 Ackermann funkcijos

3.8 apibrėžimas. *Sakysime, kad sąryšiu R(x,y), apibrėžtu aibėje A, nusakome dalinę tvarką joje, jei sąryšis refleksyvus, tranzityvus ir antisimetrinis, t.y. kokie bebūtų* $x, y \in A$ $(R(x,y) \& R(y,x)) \to x = y$.

Sąryšius, kuriais įvedama dalinė tvarka, žymime $\leq$. Kai kada, patikslindami, apie tvarką kurioje aibėje kalbama, žymėsime $\leq$ su indeksu, pavyzdžiui, $\leq_A$. Jei bet kurie du elementai palyginami, t.y. $R(x,y)$ apibrėžtas su bet kuriais x, y iš nagrinėjamosios aibės, tai tvarka *tiesine.*

3.10 apibrėžimas. *Tarkime aibėse A, B įvesta tiesinė tvarka. Aibės vadinamos panašiomis (žymime* $A \simeq B$*), jei yra izomorfinės kaip sutvarkytos aibės. Jos dar vadinamos to paties tipo aibėmis.*

Taigi, jei $A \simeq B$, tai egzistuoja tokia abipusiai vienareikšmė atitiktis tarp *A, B* elementų, kad kokie bebūtų $a_1, a_2 \in A$, jie ir jiems atitinkantys $b_1, b_2 \in B$ tenkina sąlygas: $a_1 \leq_A a_2$ ir $b_1 \leq_B b_2$. Fiksuodami kurią nors netuščią aibę, galime rasti daug jai panašių aibių. Iš visų galimų aibių išskirsime kai kurias, dažniausiai naudojamas matematikoje bei informatikoje skaitines aibes, ir suteiksime joms, o tuo pačiu ir visoms panašioms į jas, vardus. Tie vardai vadinami *tipais* arba *ordinalais.*

1. Tuščios aibės tipas 0.
2. Baigtinės aibės $N_n = \{0, 1, ..., n-1\}$ tipas n.
3. Natūraliųjų skaičių aibės tipas ω.
4. Sveikųjų skaičių aibės tipas π.
5. Racionaliųjų skaičių aibės tipas η.
6. Realiųjų skaičių aibės tipas λ.

Pakeitę $\leq$ į $\geq$, įvedame jau kitą tvarką, vadinama dar *dualiaja tvarka.* Jei aibės A tipas α, tai simboliu α^* žymimas dualiosios tvarkos tipas.

Apibrėšime veiksmus su tipais. Tarkime aibės A tipas α (tvarka $\leq_A$), o $B -$ tipas β (tvarka $\leq_B$).

3.11 apibrėžimas. *Tipų α, β suma (žymime $\alpha + \beta$) yra tiesinė tvarka $\leq$ aibėje $A \cup B$ nusakyta tokiu būdu:*
a) jei $x \in A$, $y \in B$, tai $x < y$,
b) *jei $x, y \in A$ ir $x \leq_A y$, tai $x \leq y$,*
c) *jei $x, y \in B$ ir $x \leq_B y$, tai $x \leq y$.*

Tipų α, β sandauga (žymėsime $\alpha \cdot \beta$) yra tiesinė tvarka $\leq$ aibėje $A \times B$ nusakyta tokiu būdu:
a) *jei $y_1 \leq_B y_2$, tai $(x_1, y_1) \leq (x_2, y_2)$,*
b) *jei $y_1 = y_2$ ir $x_1 \leq_A x_2$, tai $(x_1, y_1) \leq (x_2, y_2)$.*

Tarkime aibėje $A = \{x_0, x_1, x_2, ...\}$ tvarka $x_0 < x_1 < x_2 < ...$, o aibėje $B = \{y_0, y_1, y_2, ...\}$ tvarka $y_0 < y_1 < y_2 < ....$ Tuomet aibėje $A \times B$ bus tokia tvarka:

$$(x_0, y_0) < (x_1, y_0) < (x_2, y_0) < ... < (x_0, y_1) < (x_1, y_1) < (x_2, y_1) < ...$$

Nesunku matyti, kad a) $\alpha + 0 = 0 + \alpha = \alpha$, b) $1 + \omega = \omega$, bet $\omega + 1 \neq \omega$, c) $\omega^* \neq \omega$. $\alpha \times \alpha$ įprasta žymėti α^2.

Apibrėžiame funkcijas $B_n(a, x)$ su $a \geq 2$.

$B_0(a, x) = a + x$, $B_1(a, x) = a \cdot x$, $B_2(a, x) = a^x$.

Tai didėjančios funkcijos. $B_i(a, x) < B_j(a, x)$, kai $i < j$, pradedant kažkuriuo tai x_0. Jos tenkina tokias lygybes:
$B_1(a, 1) = a \qquad B_1(a, x + 1) = B_0(a, B_1(a, x))$
$B_2(a, 1) = a \qquad B_2(a, x + 1) = B_1(a, B_2(a, x))$

Pratęskime jas ($n \geq 2$):
$B_{n+1}(a, 1) = a \qquad B_{n+1}(a, x + 1) = B_n(a, B_{n+1}(a, x))$

Tariame, kad $B_{n+1}(a, 0) = 1$, kai $n \geq 1$. Ackermann funkcijos variantu, kai $a = 2$, vadiname $A(n, x) = B_n(2, x)$. Įvedame tiesinę tvarką tarp porų:

$$(0, 0) < (0, 1) < (0, 2) < ... < (1, 0) < (1, 1) < (1, 2) < ... < (n, 0) < (n, 1) < (n, 2) < ...$$

Jos tipas ω^2. Funkcija $A(n, x)$ aprašoma rekursija pagal tipą ω^2. Pastebėsime, kad reikšmei $A(n+1, 0)$ ankstesnė bus $A(n_1, x)$ su $n_1 \leq n$ ir bet kuriuo x. Ackermann funkcija nusakoma tokiomis lygybėmis:

$$A(0,x) = x+2$$
$$A(1,0) = 0$$
$$A(y,0) = 1 \text{ su } y \geq 2$$
$$A(y+1,x+1) = A(y, A(y+1,x)) \text{ visiems } x,y$$

Funkcija tenkina tokias savybes:
a) $A(n,x) \geq 2^x$ $(n \geq 2; x = 1,2,...)$
b) $A(n+1,x) \geq A(n,x)+1$
c) $A(n,x+1) > A(n,x)$ $(n,x = 1,2,...)$
d) $A(n+1,x) \geq A(n,x+2)$

Funkcija $h(x) = A(x,x)$ apibrėžta su bet kuriomis x reikšmėmis, todėl ji bendrarekursinė. Įrodysime, kad $h(x)$ nėra primityviai rekursinė. Naudosimes rezultatu, kad vieno argumento primityviai rekursinių funkcijų aibė gali būti apibrėžta naudojantis tik vieno argumento primityviai rekursinėmis funkcijomis.

Įvedame naujus sudėties bei iteracijos operatorius. Juos taikysime vieno argumento primityviai rekursinėms funkcijoms. Rezultatas $-$ vieno argumento primityviai rekursinė funkcija. Pritaikę sudėties operatorių funkcijoms $f(x), g(x)$, gauname $f(x)+g(x)$. Tarkime $g(x) \in PR$. Apibrėžiame naują funkciją $f(x)$ tokiu būdu: $f(0) = 0$, $f(x+1) = g(f(x))$. Sakysime, kad $f(x)$ gauta iš $g(x)$ pritaikius iteracijos operatorių ir žymėsime $I(g(x))$.

3.3 teorema. *Vieno argumento primityviai rekursinių funkcijų aibė sutampa su aibe, kuriai priklauso bazinės funkcijos $s(x)$, $q(x) = x \dot{-} [\sqrt{x}]^2$ ir kuri uždara sudėties, kompozicijos bei iteracijos atžvilgiu.*

Sakysime, kad $f(x)$ mažoruojama funkcija $h(x)$, jei $f(x) < h(x)$ pradedant kažkuriuo tai x_0 (t.y., kai $x \geq x_0$)). Parodysime, kad kiekviena vieno argumento primityviai rekursinė funkcija mažoruojama funkcija $h(x)$ ir todėl $h(x)$ nėra primityviai rekursinė. Tuo tikslu, visų pirma įrodysime, kad kokia bebūtų vieno argumento $f(x) \in PR$ galima rasti tokį n, kad $f(x)$ mažoruojama funkcija $A(n,x)$. Remiames 3.3 teorema, t.y. laikysime, kad nagrinėjamosios funkcijos gautos iš $s(x), q(x)$ pritaikius sudėties, kompozicijos bei iteracijos operatorius.

$s(x) < 2^x = A(2,x)$ $(x = 2,3,...)$
$q(x) < s(x) < 2^x = A(2,x)$

Tarkime $f(x) < A(n_1,x)$, $g(x) < A(n_2,x)$ ir $n = n_1+n_2$. Tuomet $f(x) < A(n,x)$ ir $g(x) < A(n,x)$.

$$f(x)+g(x) < 2 \cdot A(n,x) < 2 \cdot 2^{A(n,x)} \leq 2^{A(n+1,x)} \leq$$
$$\leq A(n, A(n+1,x)) = A(n+1,x+1) \leq A(n+2,x).$$

$$f(g(n)) < A(n, g(x)) < A(n, A(n+1, x)) = A(n+1, x+1) \leq A(n+2, x).$$

Panašiai gaunamas įvertis ir iteracijos atveju. Jei $f(x) < A(n, x)$, tai

$$f(n+x) < A(n, n+x) < A(n+x, n+x) = h(n+x).$$

Taigi, gavome, kad kokia bebūtų vieno argumento $f(x) \in PR$ ji mažoruojama visur apibrėžta funkcija $h(x)$, ir todėl $h(x)$ nėra primityviai rekursinė. Tuo pačiu įrodėme, kad aibė PR yra griežtas aibės BR poaibis.

3.8 Universaliosios funkcijos

3.12 apibrėžimas. *Tarkime A kuri nors n argumentų funkcijų aibė. Funkcija $F(x_0, x_1, ..., x_n)$ vadinama aibės A universaliąja, jei $A = \{F(0, x_1, ..., x_n), F(1, x_1, ..., x_n), ...\}$. T.y. $F(i, x_1, ..., x_n) \in A$ $(i = 0, 1, 2, ...)$ ir kokia bebūtų $f(x_1, ..., x_n) \in A$ atsiras bent vienas toks natūralusis i, kad $f(x_1, ..., x_n) = F(i, x_1, ..., x_n)$.*

Tarkime A kuri nors visur apibrėžta n argumentų funkcijų aibė, o $F(x_0, x_1, ..., x_n)$ – jos universalioji. Pastebėsime, kad jei $g(x_1, ..., x_n) = F(x_1, x_1, x_2, ..., x_n) + 1$ priklauso aibei A, tai, kadangi F universalioji, atsiras toks i, su kuriuo galioja lygybės:

$F(i, x_1, x_2, ..., x_n) = F(x_1, x_1, x_2, ..., x_n) + 1$
$F(i, i, x_2, ..., x_n) = F(i, i, x_2, ..., x_n) + 1$

Matome, kad jei $F \in PR$, tai ir $g \in PR$. Jei $F \in BR$, tai ir $g \in BR$. Iš čia išplaukia du teiginiai:

a) *visų n argumentų primityviai rekursinių funkcijų aibei universaliaja negali būti primityviai rekursinė funkcija,*

b) *visų n argumentų bendrarekursinių funkcijų aibės universaliaja negali būti bendrarekursinė funkcija.*

3.4 teorema. *Visų vieno argumento primityviai rekursinių funkcijų aibei egzistuoja universalioji bendrarekursinė funkcija.*

Įrodymas. Sutinkamai su 3.3 teorema visas vieno argumento primityviai rekursines funkcijas galima gauti iš bazinių $s(x), q(x)$ taikant sudėties, kompozicijos bei iteracijos operacijas. Funkcijoms priskirsime natūraliuosius skaičius, t.y. apibrėžiame funkcijų numeraciją. Funkcijos $f(x)$ numerį žymėsime $n(f(x))$ arba rašysime $f_n(x)$, kai jos numeriu yra n.
Funkcijoms $s(x), q(x)$ priskiriame tokius numerius: $n(s(x)) = 1$, $n(q(x)) = 3$.

Tarkime $n(f(x)) = a$, o $n(g(x)) = b$. Tuomet, funkcijoms, gautoms pritaikius sudėties, kompozicijos ar iteracijos operatorius, priskiriame tokius numerius:

$n(f(x) + g(x)) = 2 \cdot 3^a \cdot 5^b$,
$n(f(g(x))) = 4 \cdot 3^a \cdot 5^b$,
$n(I(f(x))) = 8 \cdot 3^a$.

Pavyzdžiui, $n(I(2 \cdot s)) = n(I(s+s)) = 8 \cdot 3^{2 \cdot 3 \cdot 5}$, $n(s + I(q)) = 2 \cdot 3 \cdot 5^{8 \cdot 3^3}$.

Apibrėžiame dviejų argumentų funkciją $F(n,x) = f_n(x)$ t.y. $F(n,x)$ lygi vieno argumento funkcijai su numeriu n.

$$F(n,x) = \begin{cases} f_a(x) + f_b(x), & jei \quad n = 2 \cdot 3^a \cdot 5^b \\ f_a(f_b(x)), & jei \quad n = 4 \cdot 3^a \cdot 5^b \\ f_a(f_n(x-1)), & jei \quad n = 8 \cdot 3^a, \quad x > 0 \\ 0, & jei \quad n = 8 \cdot 3^a, \quad x = 0 \\ q(x), & jei \quad n = 3 \\ s(x), & jei \quad n = 1 \end{cases}$$

Iš apibrėžimo matome, kad kiekviena funkcija turės numerį, bet ne vienintelį. Pavyzdžiui, nors $f(x){+}g(x){=}g(x){+}f(x)$, bet jų numeriai, bendru atveju, skirtingi. Ne kiekvienam natūraliajam atitinka kuri nors funkcija. Pavyzdžiui, nėra tokios funkcijos, kurios numeris lygus 7,13 ar 17. Dabar galime apibrėžti ir universaliąją $D(n,x)$ vieno argumento primityviai rekursinėms funkcijoms:

$$D(n,x) = \begin{cases} F(n,x), & \text{jei n yra kurios nors funkcijos numeris} \\ 0, & \text{priešingu atveju} \end{cases}$$

$D(n.x) -$ bendrarekursinė funkcija. Teorema įrodyta.

3.5 teorema. *$D(x_0, \alpha_n(x_1,..,x_n))$ yra visų n argumentų primityviai rekursinių funkcijų aibės universalioji.*

Įrodymas. Universaliąją žymėsime $D^{n+1}(x_0, x_1, ..., x_n)$ ir parodysime, kad ji lygi $D(x_0, \alpha_n(x_1, ..., x_n))$. Su kiekvienu fiksuotu x_0 funkcija $D(x_0, \alpha_n(x_1, ..., x_n))$ primityviai rekursinė. Antra vertus, jei $g(x_1, ..., x_n)$ kuri nors n argumentų primityviai rekursinė, tai tokia bus ir $f(x) = g(\pi_n^1(x), ..., \pi_n^n(x))$. Ji vieno argumento. Todėl atsiras toks natūralusis x_0, kad $f(x) = D(x_0, x)$. Skaičius x_0 ir bus $g(x_1, ..., x_n)$ numeriu, nes

$$f(\alpha_n(x_1, ..., x_n)) = g(\pi_n^1(\alpha_n(x_1, ..., x_n)), ..., \pi_n^n(\alpha_n(x_1, ..., x_n))) = g(x_1, ..., x_n).$$

Teorema įrodyta.

Dabar aprašysime universaliąsias dalinai rekursinėms funkcijoms. Jos bus taip pat dalinės funkcijos.

3.13 apibrėžimas. *Dalinai rekursinės funkcijos* $f(x_1, ..., x_n)$ *grafiku vadinsime aibę* $A = \{(x_1, ..., x_n, y) : f(x_1, ..., x_n) = y\}$. *Niekur neapibrėžtos funkcijos grafikas tuščia aibė.*

Pavyzdžiui, funkcijos $y = x^2$ grafikas yra aibė $\{(0, 0), (1, 1), (2, 4), (3, 9), ...\}$.

3.6 teorema. n *argumentų dalinai rekursinių funkcijų aibei egzistuoja universalioji funkcija.*

Įrodymas. Bet kurios dalinai rekursinės funkcijos grafikas − rekursyviai skaičioji aibė, nes sutampa su $\bar{sg}|f(x_1, ..., x_n) - y| - 1$ apibrėžimo sritimi. Taigi, kokia bebūtų dalinai rekursinė funkcija $f(x_1, ..., x_n)$, remiantis 3.7 apibrėžimu galime tvirtinti, kad egzistuoja tokia primityviai rekursinė funkcija $g(x_1, ..., x_n, y, z)$, kad

$(x_1, ..., x_n, y) \in A$ *tada ir tiktai tada, kai atsiras toks* z*, kad* $g(x_1, ..., x_n, y, z) = 0$

Tarkime $t = \alpha_2(y, z)$. Tuomet $(x_1, ..., x_n, y) \in A$ tada ir tiktai tada, kai atsiras toks t, kad $g(x_1, ..., x_n, \pi_2^1(t), \pi_n^2(t)) = 0$. Pažymėkime $g(x_1, ..., x_n, \pi_2^1(t), \pi_2^2(t))$ nauja funkcija $F(x_1, ..., x_n, t)$. Gavome, kad kokia bebūtų dalinai rekursinė funkcija $f(x_1, ..., x_n)$ atsiras tokia primityviai rekursinė $F(x_1, ..., x_n, t)$, kad

$$f(x_1, ..., x_n) = \pi_2^1(\mu_t(F(x_1, ..., x_n, t) = 0)) \tag{3.1}$$

Universalioji $\tilde{D}^{n+1}$ dalinai rekursinėms n argumentų funkcijoms gaunama tokiu būdu:

$$\tilde{D}^{n+1}(x_0, x_1, ..., x_n) = \pi_2^1(\mu_t(D^{n+2}(x_0, x_1, ..., x_n, t) = 0))$$

Iš tikrųjų, su kiekvienu fiksuotu x_0 $\tilde{D}^{n+1}$ dalinai rekursinė. Antra vertus, jei $f(x_1, ..., x_n)$ yra kuri nors dalinai rekursinė funkcija, tai atsiras tokia primityviai rekursinė $F(x_1, ..., x_n, t)$, kuriai galioja lygybė (3.1). Tarkime jos numeris i. Tuomet

$$\tilde{D}^{n+1}(i, x_1, ..., x_n) = \pi_2^1(\mu_t(D^{n+2}(i, x_1, ..., x_n, t) = 0)).$$

Teorema įrodyta.

3.14 apibrėžimas. *Sakysime, kad visur apibrėžta funkcija* $g(x_1, ..., x_s)$ *yra* **dalinės funkcijos** $f(x_1, ..., x_s)$ **pratęsimas,** *jei bet kuriems* $x_1^0, ..., x_s^0$*, su kuriais* f *apibrėžta, galioja lygybė* $g(x_1^0, ..., x_s^0) = f(x_1^0, ..., x_s^0)$.

Ar galima pratęsti kiekvieną dalinai rekursinę funkciją? T.y. ar arsiras tarp bendrarekursinių tokia, kuri bus jos pratęsimas? Pasirodo, kad ne visas dalinai rekursines funkcijas galima pratęsti.

3.7 teorema. *Universalioji* $\tilde{D}^{s+1}(x_0, x_1, ..., x_s)$ *dalinai rekursinėms s argumentų funkcijoms, neturi pratęsimo.*

Įrodymas. Nagrinėjame $V(x) = \bar{sg}\tilde{D}^{s+1}(x, x, ..., x)$. Jei $V(x)$ apibrėžta su kuriuo nors x_0, tai jos reikšmė lygi 1 arba 0. Tarkime $V(x)$ turi pratęsimą $W(x)$. Į ją (vieno argumento) galima žiūrėti kaip į s argumentų funkciją

$W(x_1) = pr_s^1(W(x_1), x_2, ..., x_s)$.

Atsiras toks a, kad $\tilde{D}^{s+1}(a, x_1, ..., x_s) = W(x_1)$. Ji visur apibrėžta. Imame $x_1 = ... = x_s = a$. $W(x)$ yra ir $\bar{sg}\tilde{D}^{s+1}(x, x, ..., x)$ pratęsimas. Gauname prieštaravimą $W(a) = \bar{sg}W(a)$. Taigi $V(x)$ neturi pratęsimo.
Tarkime, kad $\tilde{D}^{s+1}(x_0, x_1, ..., x_s)$ turi pratęsimą $P(x_0, x_1, ..., x_s)$. Tuomet $\bar{sg}P(x, x, ..., x)$ būtų $V(x)$ pratęsimas, o tokios tarp bendrarekursinių funkcijų nėra. Teorema įrodyta.

3.8 teorema. *Egzistuoja rekursyviai skaičiosios, bet nerekursinės aibės.*

Įrodymas. Nagrinėjame universaliąją vieno argumento funkcijoms $\tilde{D}^2(x_1, x_2)$. Funkcija $V(x) = \bar{sg}\tilde{D}^2(x, x)$ tenkina savybes:
1. $V(x)$ dalinai rekursinė,
2. $V(x)$ neturi pratęsimo,
3. $V(x)$ reikšmių aibė $\{0, 1\}$.

Lygties $V(x) = 0$ sprendinių aibė rekursyviai skaičioji, nes sutampa su dalinai rekursinės $\mu_z(V(x) + z = 0)$ apibrėžimo sritimi. Jei ji būtų rekursinė, t.y. atsirastų tokia bendrarekursinė $\kappa(x)$, kad

$$\kappa(x) = \begin{cases} 1, & \text{jei} \quad V(x) = 0 \\ 0, & \text{priešingu atveju} \end{cases}$$

tai $\bar{sg}\ \kappa(x)$ būtų $V(x)$ pratęsimas. O tai prieštarauja antrai funkcijos $V(x)$ savybei. Teorema įrodyta.

3.9 Kanoninis Post skaičiavimas

Šiame skyrelyje trumpai susipažinsim su amerikiečių logiko E.L.Post 1943 metais aprašytu *algoritmiškai apskaičiuojamųjų funkcijų* formalizmu − *kanoniniu skaičiavimu.*

3.15 apibrėžimas. *Kanoniniu skaičiavimu vadinsime ketvertą* (A, P, Ak, T). A, Ak, P, T − *baigtinės aibės. A vadinama skaičiavimo abėcėle, P - skaičiavimo kintamųjų aibė* ($A \cap P = \emptyset$) *Ak − žodžių abėcėlėje A aibė, vadinama skaičiavimo aksiomų aibe, T − taisyklių aibė pavidalo*

$$\frac{\begin{array}{c} G_{1,1}p_{1,1}G_{1,2}p_{1,2}...G_{1,n_1}p_{1,n_1}G_{1,n_1+1} \\ G_{2,1}p_{2,1}G_{2,2}p_{2,2}...G_{2,n_2}p_{2,n_2}G_{2,n_2+1} \\ ... \\ G_{m,1}p_{m,1}G_{m,2}p_{m,2}...G_{m,n_m}p_{m,n_m}G_{m,n_m+1} \end{array}}{G_1p_1G_2p_2...G_np_nG_{n+1}}$$

$G_{i,j}$ – *žodžiai abėcėlėje* A, $p_{i,j}$ – *kintamieji.*

Žodžiai virš brūkšnio vadinami taisyklės prielaidomis, o brūkšnio apačioje – išvada. Laikoma, kad kintamieji, įeinantys į išvadą, sutinkami bent vienoje prielaidoje.

Taisyklę *realizuojančiu rinkiniu* vadinsime reiškinį pavidalo

$$\begin{pmatrix} p^1, & p^2, & ..., & p^s \\ B_1, & B_2, & ..., & B_s \end{pmatrix}$$

Čia p^1,...,p^s – pilnas sąrašas kintamųjų, įeinančių į taisyklę, o $B_1, ..., B_s$ – kurie nors žodžiai abėcėlėje A. Pakeitę taisyklėje visas p^i įeitis žodžiais A_i $(i = 1, ..., s)$, gauname taisyklės taikymą

$$\frac{\begin{array}{c} Q_1 \\ \vdots \\ Q_m \end{array}}{Q}$$

Q, Q_i $(i = 1, ..., m)$ – žodžiai abėcėlėje A. Žodžių abėcėlėje A seka vadinama išvedimu, jei kiekvienas jos narys yra aksioma arba gautas iš kairėje esančių formulių pritaikius kurią nors skaičiavimo taisyklę. Sakoma, kad žodis B išvedamas skaičiavime, jei galima rasti išvedimą, kuris baigiasi žodžiu B.

Kanoninis skaičiavimas įdomus tuo, kad savyje apjungia kaip Turing mašinas, taip ir loginius skaičiavimus. Į Turing mašinas ir loginius skaičiavimus galime žiūrėti kaip į atskirus kanoninių skaičiavimų atvejus. Gauname ir kitokį bendresnį formalų aparatą rekursyviai skaičiosioms aibėms aprašyti. Generuojami objektai nebūtinai yra skaičiai.

Pavyzdžiai.
1. $A = \{|\}$, $P = \{p\}$, $Ak = \{||\}$, $T = \frac{p}{pp}$.

Išvedamų skaičiavime žodžių aibė lygi $\{||, ||||, ..., |^{2^n}, \}$.

2. $A = \{1, 0, *\}$, $P = \{p, q\}$, $Ak = \{B\}$ B – kuris nors žodis abėcėlėje $\{1, 0\}$. T susideda iš taisyklių:

$$\frac{p}{p*} \qquad \frac{p1*q}{p*0q} \qquad \frac{p0*q}{p*1q} \qquad \frac{*p}{p}$$

Kai kada domina ne visi išvedami žodžiai abėcėlėje A, o išvedami žodžiai abėcėlėje A', kuri yra kuris nors aibės A poaibis. Tuo atveju sakoma, kad A' yra pagrindinė skaičiavimo abėcėlė. Jei antrajame pavyzdyje pagrindine abėcėle laikysime $\{1, 0\}$, tai skaičiavime išvedamas tik vienas (neskaitant aksiomos) žodis, kuris gaunamas iš B, pakeitus jame visas 0 įeitis vienetukais, bei 1 įeitis nuliukais.

3.16 apibrėžimas. *Sakysime, kad du skaičiavimai ekvivalentūs atžvilgiu pagrindinės abėcėlės A, jei išvedamų abejuose skaičiavimuose žodžių abėcėlėje A aibės sutampa.*

3.17 apibrėžimas. *Taisyklę, kurioje bent vieno kintamojo, įeinančio į prielaidą, nėra išvadoje, vadinsime c-taisykle.*

3.4 lema. *Koks bebūtų kanoninis skaičiavimas $\Pi = (A, P, Ak, T)$, galima rasti jam ekvivalentų atžvilgiu pagrindinės abėcėlės A, kuriame nėra c-taisyklių.*

Įrodymas. Tarkime $A = \{a_1, a_2, ..., a_n\}$ ir skaičiavimo Π taisyklėje $G_1, ..., G_m/G$ kintamųjų $p_1, ..., p_s$, įeinančių į prielaidas, nėra išvadoje G. Naujasis skaičiavimas, kuriame bus eliminuota nagrinėjamojo c-taisyklė, ir kuris ekvivalentus skaičiavimui Π atžvilgiu pagrindinės abėcėlės A, gaunamas tokiu būdu. Abėcėlė papildoma nauju simboliu, pavyzdžiui $*$. O taisyklė keičiama tokiomis:

$$\frac{\begin{matrix} G_1 \\ \vdots \\ G_m \end{matrix}}{p_1 ... p_s * G} \qquad \frac{pa_i * q}{p * q} \quad (i = 1, ..., n) \qquad \frac{*q}{q}.$$

Taigi, generuodami tam tikrus "tarpinius" žodžius, kuriuose yra $*$, gauname skaičiavimą, kuriame eliminuota c-taisyklė ir jis ekvivalentus skaičiavimui Π atžvilgiu pagrindinės abėcėlės A. Lema įrodyta.

3.18 apibrėžimas. *Kanoninis skaičiavimas $\Pi = (A, P, Ak, T)$ vadinamas normaliuoju, jei aibėje Ak tėra vienas elementas, o visos taisyklės yra pavidalo*

$$\frac{Gq}{qG'}$$

G, G' – žodžiai abėcėlėje A.

Parodysime, kaip galima modeliuoti Turing mašinos darbą su fiksuotais pradiniais duomenimis. Tuo tikslu nagrinėsime determinuotą standartinę Turing mašiną su vienpuse viena juosta. Tarkime, Turing mašinos abėcėlė $A \cup \{b\}$, $A = \{a_1, ..., a_m\}$. Būsenų aibė $Q = \{q_0, q_1, ..., q_s\}$. q_0 – pradinė būsena. Pradiniai duomenys $e_1 e_2 ... e_v$ – žodžiai abėcėlėje A. Jie užrašomi pirmose (iš kairės į dešinę) v ląstelėse. Po baigtinio

žingsnių skaičiaus ji pereina į galutinę būseną (pažymėsime ją q_s) arba dirba be galo ilgai. Jei baigia darbą, tai pereina į galutinę būseną q_s ir skaitančioji galvutė bus ties pirmaja ląstele. Tuščios ląstelės gali būti tik dešinėje galutinio rezultato t.y. tuščios ląstelės prirašomos tik iš dešinės. Skaičiavimo eigoje jų negali būti tarp abėcėlėje A žodžių.
Nors iš pirmo žvilgsnio ir atrodo, kad Turing mašina turi tenkinti daug apribojimų, bet, yra žinoma, kad kokia bebūtų Turing mašina, galima rasti jai ekvivalenčią, tenkinančią išvardintus apribojimus.

Jos darbą modeliuojantis normalusis kanoninis skaičiavimas gaunamas tokiu būdu. Abėcėlė $B = \{a_1, ..., a_m, q_0, q_1, ..., q_s, b, *\}$. Pagrindinė abėcėlė $A \subset B$. $P = \{p\}$, $Ak = \{*q_0e_1e_2...e_v\}$. Taisyklės:

$$\frac{xp}{px} \quad (x \in B) \qquad \frac{*q_sp}{p**} \qquad \frac{b**p}{p**} \qquad \frac{x**p}{p} \quad (x \in A)$$

Kiekvienai mašinos komandai atitiks po taisyklę. Komandoms pavidalo $\delta(q_i, a_j) = (q_u, a_v, D)$, $\delta(q_i, b) = (q_u, y, D)$ $(y \in A \cup \{b\})$, $\delta(q_i, a_j) = (q_u, a_v, K)$, $\delta(q_i, y) = (q_u, z, N)$ $(y, z \in A \cup \{b\})$ priskiriame taisykles:

$$\frac{q_ia_jp}{pa_vq_u} \qquad \frac{q_i*p}{pyq_u*} \qquad \frac{xq_ia_jp}{pq_jxa_v} \quad (x \in A) \qquad \frac{q_iyp}{pq_uz}$$

Kita teorema priklauso logikui E.L.Post.

3.9 teorema. *Koks bebūtų kanoninis skaičiavimas su pagrindine abėcėle A, galima rasti jam ekvivalentų normalųjį atžvilgiu A.*

3.10 Pratimai

1. Įrodyti, kad funkcija primityviai rekursinė:

a) $x \cdot y$

b) x^y

c) $x \dot{-} 1$

d) $n!$, kai $0! = 1$

2. Duota, kad $g(x_1, ..., x_n)$ primityviai rekursinė. Įrodyti, kad funkcija f irgi primityviai rekursinė:

a)

$$f(x_1, ..., x_n) = \prod_{i=0}^{x_n} g(x_1, ..., x_{n-1}, i),$$

b)

$$f(x_1, ..., x_{n-1}, y, z) = \begin{cases} \sum_{i=y}^{z} g(x_1, ..., x_{n-1}, i), & jei \quad y \leq z \\ 0, & jei \quad y > z \end{cases}$$

3. Duota, kad n argumentų funkcijos *f,k,h* primityviai rekursinės. Įrodyti, kad

$$f(x_1, ..., x_n) = \sum_{i=h(x_1,...,x_n)}^{k(x_1,...,x_n)} g(x_1, ..., x_{n-1}, i)$$

primityviai rekursinė.

4. Įrodyti, kad funkcija

$$div(x, y) = \begin{cases} 1, & \text{jei } x \text{ dalosi iš } y \\ 0, & \text{priešingu atveju} \end{cases}$$

primityviai rekursinė.

5. Įrodyti, kad funkcija $nd\ \ x$, kurios reikšmė lygi x daliklių (įskaitant ir vienetą) skaičiui, primityviai rekursinė.

6. Parašyti pirmuosius tris ketvertus pagal Cantor numeraciją.

7. Duota rekursinė aibė A. Įrodyti, kad jos papildinys taip pat rekursinė aibė.

8. Duotos rekursinės aibės $A_1, ..., A_n$. Įrodyti, kad jų sąjunga bei sankirta taip pat rekursinės aibės.

9. Įrodyti, kad jei $f(x) \in PR$, tai lygties $f(x) = 0$ sprendinių aibė rekursinė.

10. Ar $\pi^* = \pi$?

11. Kam lygus $\omega^* + \omega$ tipas ?

12. Nustatyti aibės tipą
$(0,0) < (0,1) < (0,2) < ... < (1,0) < (1,1) < (1,2) < ...$

13. Kokią funkciją apibrėžia $B_3(a, n)$?

14. Įrodyti, kad iš bazinių funkcijų $s(x), q(x)$, naudojantis sudėties, kompozicijos bei iteracijos operatoriais, galima gauti funkciją: a) $pr_1^1(x)$, b)$f(x) \equiv 0$, c) $sg\ \ x$.

15. Kam lygi funkcija: a) $I(x + 2 \cdot \sqrt{x} + 1)$, b) $I(\bar{sg}x)$?

16. Rasti funkcijų, aprašytų 14-toje užduotyje, numerius.