<u>Mövzu: Primitiv rekursiv funksiyalar.</u>

<u>Plan</u>

1. Hesablama funksiya
2. Rekursiv funksiya
3. Primitiv- rekursiv funksiya

1) $x_1, x_2, ..., x_n$ dəyişənlərinin ixtiyari qiymətlərində funksiyanın y qiymətini təyin etməyə imkan verən alqoritm varsa, $y = \varphi(x_1, x_2, ..., x_n)$ funksiyası hesablanan funksiya adlanır.

Hesablama alqoritmi anlayışı dəqiq təyin edilmədiyindən bu tərif də intuitiv qəbul edilməlidir.

Həmin tərifi dəqiqləşdirməküçün ən sadə elementar funksiyalardan başlayaraq, hesablanan funksiyalar qurmağa cəhd edəcəyik.

Mənfi olmayan tam ədədlərdən və dəyişənlərdən toplama, hesabi çıxma (burada $|x - y|$ kimi başa düşülür), vurma, hesabi bölmə (burada $[a/b] = ant(a/b)$ kimi başa düşülür), cəmlər və hasil qurulmasının köməyi ilə alınan funksiyaları elementar funksiyalar adlandıracağıq.

Elementar funksiyaların qurulmasında istifadə bütün əməliyyatların yerinə yetirirlməsi üçün alqoritmlər yaxşı məlum olduğundan elementar funksiyaların hesablanan olması şübhə doğurmur.

Elementar funksiyaların qurulması üçün başlanğıc kimi təkcə 1 ədədini götürmək olar. Doğrudan da,

$$0 = |1 - 1|,\ 2 = 1 + 1,\ 3 = (1 + 1) + 1,\ \ldots$$

Elementar funksiyalara misallar göstərək:

1. Aşağıdakı şəkillərdə olan bütün sadə funksiyalar elementar funksiyalardır:

$\varphi(x) \equiv x+1,\ \varphi(y) \equiv 12y,\ \varphi(a,b,c) \equiv ab+c,\ \xi(b) \equiv b^2\ \ (b^2 = b \cdot b$ olduğundan) və s.

2. Ədədlər nəzəriyyəsində tez- tez istifadə olunan funksiyaların əksəriyyəti elementar funksiyalardır. Məsələn,

a) $\min(x,y) = [(|x+y| - |x-y|)/2]$

b) $\operatorname{sgn}(x) = \begin{cases} 1, & x \geq 1 \ \ olduqda \\ 0, & x = 0 \ \ olduqda \end{cases}$ $\quad \operatorname{sgn} x = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$

bu funksiyanı $\min(x,y)$ funksiyası vasitəsilə

$$\operatorname{sgn}(x) = \min(x,1)$$

şəklində ifadə etmək olar.

c) $x \leq y$ bərabərsizliyi

$$\overline{\operatorname{sgn}}\, x = |1 - \operatorname{sgn} x| = \begin{cases} 0, & x \geq 0 \quad olduqda \\ 1, & x = 0 \quad olduqda \end{cases}$$

$$\min(x,y) = x \text{ və ya } |\min(x,y) - x| = 0$$

bərabərliyi ilə eynigüclüdür.

Onda “x kiçikdir və ya böyükdür” predikatını

$$P(x,y) = \overline{\operatorname{sgn}}(|\min(x,y) - x|)$$

şəklində yazmaq olar.

Doğrudan da konkret x^* və y^* qiymətləri üçün $x^* \leq y^*$ olduqda $P(x^*, y^*) = 1$ olur, yəni predikat doğru bərabərliyə çevrilir, əks halda P=0 olur.

d) Bizə

$$y - x = \begin{cases} |y - x|, & y \geq x \quad olduqda \\ 0, & y < x \quad olduqda \end{cases}$$

şəklində təyin olunmuş funksiya da lazım olacaqdır. Bu funksiyanı

$$y-x=|y-x|\overline{\operatorname{sgn}}(|\min(x,y)-x|)$$

şəklində göstərmək olar. Deməli *y*-*x* funksiyası da elementar funksiyadır.

e) *a*-nın *n*-ə bölünməsindən alınan qalıq

$res(a,n)=|a-n[a/n]|$ - də elementar funksiyadır.

Belə bir sual meydana çıxır: bütün hesablanan funksiyalar sinifi bütün elementar funksiyalar sinifindən genişdirmi? Elementar olmayan hesablanan funksiya qurmaq olarmı?

Elementar funksiyalar içərisində ən yüksək sürətlə artan funksiyanın hasil (vurma) olması aşkardır. $a\cdot n$ hasili $a\cdot n=\underbrace{a+a+\ldots+a}_{n}$ toplamanın iterasiyasıdır.

$$a^n=a\cdot a\cdot\ldots\cdot a=\prod_1^n a$$

Bu funksiya hasil vasitəsilə ifadə olunduğu üçün (hələ) elementar funksiyadır, amma *a*-nın və *n*-nin artması ilə çox sürətlə artır.

Daha sürətlə artan funksiya qurmaq üçün qüvvətə yüksəltmənin iterasiyasına baxaq.

$$\psi(0,a)=a,\quad \psi(1,a)=a^a,\quad \psi(2,a)=a^{\left(a^a\right)},$$

və ümumiyyətlə

$$\psi(n+1,a)=a^{\psi(n,a)} \qquad (1)$$

Bu funksiya fövqəlada sürətlə artır. İsbat etmək olar ki, elementar funksiyalar qurmaq vasitəsilə $\psi(n,a)$ funksiyasının artım sürətinə “çatmaq, ona yetişmək” mümkün deyildir. Daha dəqiq desək, *a*-nın müəyyən a^* qiymətindən sonra $\psi(n,a)$ bütün elementar funksiyaları majorantlayır, yəni ixtiyari $\varphi(a)$ elementar funksiyası üçün elə m^* tapılır ki, $a \geq a^*$ bərabərsizliyini ödəyən bütün a-lar üçün

$$\varphi(a) < \psi\left(m^*, a\right)$$

bərabərsizliyi ödənilir. Onda isbat etmək olar ki, $\gamma(n) = \psi(n, n)$ elementar funksiya deyildir.

Beləliklə, qüvvətə yüksəltmə əməli qeyri- elementar funksiya qurmağa imkan verir. Amma $\psi(n, a)$ funksiyası hesablanandır.

2) Biz rekursiv funksiyalar sinifini intuitiv təyin edəcəyik. Bunun üçün əvvəlcə rekursiv funksiyaları sadalayacaq və sonra həmin funksiyalardan yenilərini düzəltmə qaydalarını göstərərcəyik.

<u>Tərf 1.</u> I Aşağıdakı funksiyalar ilk rekursiv funksiyalar adlanır.

a) sıfır funksiya

$$\forall x \in N; \;\; Z(x) = 0\,;$$

b) vahid əlavə etmə funksiyası:

$$\forall x \in N; \;\; N(x) = x + 1\,;$$

v) proyektləndirici funksiyalar:

$$\forall x_i \in N \;\left(i = \overline{1, n}\right); \quad U_1^n(x_1, x_2, ..., x_n) = x_i \quad (i = 1,2,3,...)$$

II. Baxılan funksiyalardan aşağıdakı qaydalar əsasında yeni funksiyalar düzəldilir:

q) Əvəzetmə (yerləşdirmə) qaydası:

f funksiyası $g(y_1, y_2, ..., y_n)$ və $h_1(x_1, x_2, ..., x_n)$, $h_2(x_1, x_2, ..., x_n)$,..., $h_1(x_1, x_2, ..., x_n)$ funksiyalarından

$$f(x_1, x_2, ..., x_n) = g(h_1(x_1, x_2, ..., x_n), ..., h_m(x_1, x_2, ..., x_n))$$

şəklində alınır;

ğ) Rekursiya qaydası: *f* funksiyası *g* və *h* funksiyalarından

$$f(x_1,....,x_n,0)=g(x_1,...,x_n)$$

$$f(x_1,....,x_n,y+1)=h(x_1,...,x_n,y,f(x_1,...,x_n,y))$$

rekursiyalarının köməyi ilə alınır; burada $x_1,x_2,..,x_n$ rekursiyalar parametrləridir və $n\neq 0$; bundan başqa, $n=0$ olduqda $f(0)=k$ (k- qeyd olunmuş ədəddir) və $f(y+1)=h(y,f(y))$

d) μ - operator qaydası (minimumlaşdırma qaydası) y-in

$$g(x_1,x_2,...,x_n,y)=0 \qquad (*)$$

tənliyini ödəyən ən kiçik qiymətini $\mu y(g(x_1,x_2,...,x_n,y)=0)$ ilə işarə etdikdə,

$$f(x_1,x_2,...,x_n)=\mu y(g(x_1,x_2,...,x_n,y)=0)$$

olarsa və ixtiyari $x_1,x_2,...,x_n$ üçün y-in (*) münasibəti ödəyən heç olmasa bir qiyməti varsa, deyirlər ki, *f* funksiyası g funksiyasından μ - operator vasitəsilə alınmışdır.

Tərif 2. *f* funksiyası ilk rekursiv funksiyalardan əvəzetmə, rekursiya və μ - operator qaydalarının sonlu sayda tətbiq edilməsi nəticəsində alındıqda *f*-ə rekursiv (ümumrekursiv) funksiya deyilir.

Xüsusi halda, rekursiv funksiya ancaq əvəzetmə və rekursiya qaydalarının tətbiqi ilə alınmışdırsa, həmin funksiya primitiv-rekursiv funksiya adlanır.

Beləliklə, hər ir primitiv-rekursiv funksiya rekursiv funksiyalar; amma bunun tərsi doğru deyildir. Başqa sözlə, rekursiv funksiyalar a)-v) bəndlərində göstərilən ilk funksiyalardan ibarət olan və q)- d) qaydaalarına nəzərən qapalı funksiyalar sinifidir.

Göstərək ki, rekursiv funksiyadan fiktiv (saxta) dəyişənlərin daxil edilməsi dəyişənlərin yerdəyişməsi və eyniləşdirilməsi nəticəsində də rekursiv funksiya alınır. Başqa sözlə aşağıdakı teorem doğrudur.

Teorem 2. Əgər $g(y_1, y_2,..., y_n)$ rekursiv funksiyadırsa, $x_1, x_2,..., x_n$ müxtəlif dəyişənlərdirsə və $\forall i \ \left(i = \overline{1,k}\right)$ üçün z_i dəyişəni $x_1, x_2,..., x_n$ dəyişənlərindən biridirsə

$$f(x_1, x_2,..., x_n) = g(z_1, z_2,..., z_k)$$

funksiyası da rekursivdir.

Teorem 3. (Teorem 2-nin nəticəsi)unksiyası

a) $z(x_1, x_2,..., x_n) = 0$ sıfır funksiyası primitiv rekursivdir.

b) $C_n^k(x_1, x_2,..., x_n) = k$ - sabit funksiya primitiv rekursivdir.

v) əvəzetmə qaydası h_i $x_1, x_2,..., x_n$ dəyişənlərindən ancaq bəzilərindən asılı olan hal üçün genişləndirilə bilərş Analoji olaraq, rekursiv qaydası g funksiyası $x_1, x_2,..., x_n$ dəyişənlərindən h funksiyası isə $x_1, x_2,..., x_n, y,$ $f(x_1, x_2,..., x_n)$ dəyişənlərinin ancaq bəzilərindən asılı olan hal üçün genişləndirilə bilər.

Teorem 4. Aşağıdakı funksiyalar primitiv rekursivdir.

1) x+y

2) $x \cdot y$

3) x^y

4) $\delta(x) = \begin{cases} x^{-1}, & x > 0 \ \ olduqda \\ 0, & x = 0 \ \ olduqda \end{cases}$

5) $x \overset{\bullet}{\rightarrow} y = \begin{cases} x - y, & x > y \ \ olduqda \\ 0, & x \le y \ \ olduqda \end{cases}$ (hesabi çıxma)

6) $|x-y| = \begin{cases} x-y, & x \geq y \ \ olduqda \\ y-x, & x < y \ \ olduqda \end{cases}$

7) $\operatorname{sgn}(x) = \begin{cases} 0, & x = 0 \ \ olduqda \\ 1, & x \neq 0 \ \ olduqda \end{cases}$

8) $\overline{\operatorname{sgn}}(x) = \begin{cases} 1, & x = 0 \ \ olduqda \\ 0, & x \neq 0 \ \ olduqda \end{cases}$

9) $x!$ 10) $\min(x, y)$ 11) $\min(x_1, x_2, ..., x_n)$

12) $\max(x, y)$ 13) $\max(x_1, x_2, ..., x_n)$

14) $r(t, x) = \begin{cases} y - in\ x - \text{ə bölünməsindən } alınan\ qalığa, x \neq 0 \\ 0, \quad x = 0 \ \ olduqda \end{cases}$

15) $pt = y$ - in x-ə bölünməsindən alınan qismətə

$$\operatorname{sgn} x = \begin{cases} 1, & \text{əgər } x > 0 \\ 0, & \text{əgər } x = 0 \\ -1, & \text{əgər } x < 0 \end{cases}$$

Mövzu: **<u>Primitiv rekursiv operatorlar. Akkerman funksiyası</u>**

<u>Plan</u>

1. Primitiv rekursiv funksiyalar
2. Akkerman funksiyası

Operator primitiv rekursiv (PR) adlanır, əgər funksiyanın primitiv rekursivliyini saxlayırsa.

S_m^n və R_n PR-dir, tərifə görə.

$h(y_1,...,y_m), g(x_1,...,x_n),...,g_m(x_1,...,x_n)$ funksiyaları üçün

$$S_m^n(h, g_1,...,g_m) = h(g_1(x_1,...,x_n),....,g_m(x_1,...,x_n)) = f(x_1,...,x_n)$$

Bu tərif $\{S_m^n\}$ operatorlarının superpozisiyasını doğurur.

R_n (n+1) yerli f funksiyasını n yerli g və (n+2) yerli h funksiyası ilə belə təyin edir.

$$\begin{cases} f(x_1,...,x_n,0) = g(x_1,...,x_n) \\ f(x_1,...,x_n,y+1) = h(x_1,...,x_n,y, f(x_1,...,x_n,y)) \end{cases}$$

şərti keçid operatorun (qollara ayrılma ilə) burada B ilə işarə edək, hansı ki, $q_1(x_1,...,x_n), q_2(x_1,...,x_n)$ funksiyaları və $p(x_1, x_2,...,x_n)$ predikatı üçün $f(x_1,...,x_n) = B(q_1, q_2, p)$ funksiyasını qurur.

$$f(x_1,...,x_n) = \begin{cases} \mathrm{q}_1(x_1,...,x_2) & \text{əgər } P(x_1,...,x_n) \ \textit{olduğundan} \\ q_2(x_1,...,x_n) & \text{əgər } \mathrm{P}(\mathrm{x}_1,...,x_n) \ \textit{olduğundan} \end{cases} \qquad (1)$$

B-nin PR-liyi (1)-ə ekvivalent olan növbəti münasibətlərindən aydındır (görünür).

$$f(x_1,...,x_n) = g_1(x_1,...,x_n)\chi(x_1,...,x_n) + g_2(x_1,..,x_n)\chi(x_1,...,x_n)$$

Tutaq ki, $f(x_1,...,x_n,y)$- (n+1) dəyişənli funksiyadır. Yaxşı məlumdur ki, əməliyyatlar cəm və hasil ($\sum\limits_{y<x}$ və $\prod\limits_{y<z}$) z (limitli) sərhədli y dəyişənli, hansı ki, $f(x_1,...,x_n,y)$ funksiyasından yeni funksiyalar doğur.

$$q(x_1,...,x_n,z)=\sum_{y<z} f(x_1,...,x_n,y)$$

$$h(x_1,...,x_n,z)=\prod_{y<z} f(x_1,...,x_n,y)$$

Onlar PR-dir. (Əgər f – primitiv rekursivdir) və bu münasibətlərin köməyi ilə

$$g(x_1,...,x_n,0)=0 \qquad \text{(tərifə görə)}$$

$$g(x_1,...,x_n,z+1)=g(x_1,...,x_n,z)+f(x_1,...,x_n,z);$$

$$h(x_1,...,x_n,0)=1 \qquad \text{(tərifə görə)}$$

$$h(x_1,...,x_n,z+1)=h(x_1,...,x_n,z)\cdot f(x_1,...,x_n,z)$$

İndi asanlıqla göstərmək olar ki, $\sum\limits_{y=k}^{z}$ və $\prod\limits_{y=k}^{z}$ operatorları primitiv rekursivdir.

Ən kiçik ədədlə məhdudlanan operator (μ- operator) və ya minimallaşmış məhdud operator, hansı ki, predikatlara tətbiq olunan, belə təyin olunur:

$$\mu y_{y\le z}P(x_1,...,x_n,y)=\begin{cases}\text{ən kiçik } y\le z \ -\text{, eləsinə } ki,\\ P(x_1,...,x_n,y) \quad doğrudur,\\ \text{əgər belə } y \text{ varsa},\\ z-\text{ə əks } halda.\end{cases}$$

$P(x_1,...,x_n,y)$- predikatindan $\mu y_{y\le z}$ operatoru vasitəsilə $f(x_1,...,x_n,z)$ funksiyası alınır. İkinci hal, tərifdə μ əlavə olunur ona görə ki, f hər yerdə təyin olunsun.

Məhdud μ- operatoru PR- dir.

$$\mu y_{y\le z} P(x_1,...,x_n,y) = \sum_{i=0}^{z} \prod_{j=0}^{i} \left(1-\chi_P\left(x_1,...,x_j\right)\right)$$

Cəmləmənin z-kəmiyyətinə kimi olması məhdudluğu μ- məhdud operatorda hesablamanın qurtarmasına cavabdehlik veriv; harada ki, P predikatının yuxarıdan hesablanmasın qiymətləndirilməsi sonralar göstərəcəyik ki, qeyri- məhdud μ- operator PR deyil.

μ-operatoru (məhdud eləcə də qeyri məhdud) tərs funksiyasının qurulması üçün əlverişli üsuldur. Doğrudan da $g(x)=\mu(y)\ (f(y)=x)$ funksiyası $f(x)$- in tərsidir.

Buna görə biryerli funksiyalara tətbiq olunan μ- uperatoruna (tətbiqi) bəzi vaxt müraciət operatoru deyilir.

Misal 1. ($\sqrt{x}$) funksiyası PR- dir.

$$\sqrt{x} = \mu y_{y\le z}\left((y+1)^2 > x\right)$$

Misal 2. Loqarifmin tam hissəsi ixtiyari tam k əsasına görə primitiv rekursivdir.

$$\left[\log_k^x\right] = \mu y_{y\le z}\ \left(k^{y+1} > x\right)$$

qeyd edək. Birincisi bütün primitiv rekursiv funksiyalar hər yerdə təyin olunub. Bu oradan çıxır ki, sadə funksiyalar hər yerdə təyin olunub. S_m^n vı R_n operatorları bu xassəni saxlayır. İkincisi, ciddi desək, biz funksiyalarla işimiz var, onların primitiv- rekursiv yazılışdan. Bu yazılışı ekvivalentliyə görə siniflərə bölmək olar, eyni bir funksiyanı verən bütün yazılışları.

2) Akkerman funksiyaları.

Keçmiş bəndlərdə apardığımız araşdırmalardan sonra təbii olaraq belə bir sual meydana çıxır: bütün hesablanan funksiyaları PR – kimi yazmaq olarmı? Bunun cavabının mənfi olmasını göstərmək üçün elə misal quraq ki, hesablanan funksiya PR deyil. Misal ideyası belə ki, elə funksiyalar ardıcıllığı qurmalı ki, hər biri əvvəlkindən sürətlə yaranır (doğulur, əmələ gəlir) və konstruksiyalaşdırmaq onu elə funksiyaların köməyi ilə, harada ki, istənilən PR-dən sürətlə əmələ gəlir.

Beləliklə, biz istəyirik ki, imkan daxilində sadə, ancaq tez sürətlə yaranan funksiya tapaq. Təcrübə göstərir ki, hasil cəmdən tez yaranır (artır, böyüyür), qüvvətdə hasildən toplananı, vurmanı və qüvvətə yüksəltmənin 0-cı, 1-ci, 2-ci dərəcəli əməllər adlandırsaq və bunlar üçün birobrazlı ifadə etmək məqsədi ilə

$$P_0(a,x) = a + x,\ \ P_1(a,x) = ax,\ \ P_2(a,x) = a^x\ ,$$

hamıya məlum olan ideya ilə hansı ki, bu ardıcıllığı daha yüksək tərtibli qüvvətə yüksəltmə yolu ilə. Bunun üçün yüksək tərtibli əməllər əvvəlki əməlin üzərində yeni tərtibindən olan əməldən meydana çıxır. Belə ki, vurma toplamadan alınır, qüvvətə yüksəltmə isə hasildən alınır.

P_0, P_1, P_2 funksiyaları arasında aşağıdakı əlaqələrlə əlaqələndirmişlər:

$$\begin{aligned} &P_2(a,x+1) = P_0(a,P_1(a,x)); \quad P_1(a,1) = a;\ P_1(a,0) = 0 \\ &P_2(a,x+1) = P_1(a,P_2(a,x)); \quad P_2(a,1) = a;\ P_2(a,0) = 1 \end{aligned} \qquad (1)$$

n=2,3,... üçün

Bu zənciri davam etdirərək,

$$\begin{aligned} &P_{n+1}(a,0) = 1; \\ &P_{n+1}(a,1) = a; \\ &P_{n+1}(a,x+1) = P_0(a,P_{n+1}(a,x)) \end{aligned} \qquad (2)$$

Aşkardır ki, (1) ifadəsindən alınır və $P_2(a,0)=0$. P_n çox sürətlə artır. Məsələn n=3 üçün alarıq.

$$P_3(a,0)=1;\ P_3(a,1)=a;\ P_3(a,2)=a^a;\ P_3(a;3)=a^{a^a}$$

Yeni funksiyalar daxil edək.

$$B(n,x)=P_n(2,x),\quad A(x)=B(x,x)$$

$B(n,x)$ funksiyası Akkerman funksiyası adlanır, $A(x)$-ə Akkerman dioqanal funksiyası adlanır.

$B(n,x)$ funksiyası (2)- dən aşağıdakı eyniliklər alınır:

$$\begin{cases} B(0,x)=2+x \\ B(n+1,0)=sg(n) \\ B(n+1,x+1)=B(n,B(n+1,x)) \end{cases} \qquad (3)$$

Bu əlaqələr $B(n,x)$- in qiymətini hesablamağa imkan verib, eyni ilə $A(x)$, həm də funksiyanın həmin nöqtədə qiymətini hesablamaq üçün funksiyanın keçmiş (əvvəlki) nöqtələrdəki, qiymətinə müraciət etmək lazımdır- tamamilə PR- dəki sxem kimi. Bir halda rekursiya iki dəyişənli kimi o dəqiqə verilir, bu təbii ki, nöqtələrin sahmanlı düzülüşünə və funksiyanın əvvəlki nöqtələrdəki qiymətinin tapılmasını lazımı qədər mürəkkəbləşdirir.

Teorem. $A(x)$ Akkerman funksiyası ixtiyari PR-dən sürətlə böyüyür və deməli PR deyil. Daha dəqiq ixtiyari biryerli PR $f(x)$ funksiyası üçün elə n tapılır ki, $\forall x \geq n$, $A(x) > f(x)$.

Beləliklə, Akkerman funksiyası total (yaxud təyin olunan) hesablanan funksiyadır, belə ki, o PR deyil. Onda biz o ideyadan etiraz etməliyik ki, PR funksiyası dəqiq formal, formal olmayanın alqoritm anlayışının analoqudur.

Mövzu: **<u>Hissə- hissə rekursiv funksiyalar. Çörç tezisi</u>**.

<u>Plan</u>

1. Hissə- hissə rekursiv funksiyalar.
2. Çörç tezisi.

$A(x)$ -Akkerman ffunksiyası misalı göstərir ki, hesablanan funksiyaların qurulması üçün lazım olanların genişləndirilməsi lazım gəlir. Göstərmək olar ki, təkrarlanan rekursiyalı operatorlar (yəni eyni zamanda çoxlu sayda dəyişənli) bütün hesablanan funksiyalar sinifini istənilən kimi qapanır. Bu məqsəd üçün daha yaxşı hal qeyri- məhdud μ- operatordur (sonralar qeyri-məhdud sözünün tətbiqini buraxacayıq)

Əgər funksiya sadə $0, x+1, U_m^n$ funksiyalarından sonlu sayda superpozisiya, primitiv- rekursiv və μ- operatorların tətbiqi ilə qurularsa o hissə- hissə rekursiv funksiya adlanır.

Tərifə görə μ- operatoru predikatlara tətbiq olunur. Rekursiv funksiyalar nəzəriyyəsində $P(x)$ predikatının doğruluğu həmişə hər hansı bərabərliyin doğruluğu ilə əlaqədardır (məsələn $\chi_p(x)=0$) və tərsinə hər bir bərabərlik burada dəyişəndən asılı predikatdır, onda μ- operatoruna standart forma vermək olar, məsələn

$$f(x_1, x_2, .., x_n) = \mu y(g(x_1, ..., x_{n-1}, y) = x_n)$$

və ya

$$f(x_1, x_2, .., x_n) = \mu y(g(x_1, ..., x_n, y) = 0)$$

Gələcəkdə μ- operatoru hesablanan funksiyalara tətbiq olunduqda yenidən hesablanan funksiya verir. $(x_1, x_2, ..., x_n)$- i götürəndə

$f(x_1, x_2, .., x_n) = \mu y(g(x_1, ..., x_n, y) = 0)$ funksiyasını hesablamaq üçün sadə prosedura mövcuddur: $(x_1, ..., x_n, 0)$, $(x_1, ..., x_n, 1)$,... seçimində g-ni hesablayırıq o vaxta qədər ki, hələlik sıfır qiyməti almamışıq. Bir halda, əvvəl baxılan prosedurdan fərqli olaraq, o bu nəticəyə gətirə bilər: verilmiş $(x_1, x_2, ..., x_n)$ götürümündə $g(x_1, ..., x_n, y) = 0$ tənliyinin həlli yoxdur halında baş verir. Bu halda $g(x_1, ..., x_n, y)$ funksiyası qeyri - müəyyəndir. Məsələn $x+1$ funksiyasının tərsi olan $x-1 = \mu y \ (y+1 = x)$ $x = 0$ olduqda təyin olunmayıb. Qeyd edək ki, burada təyin olunmama mexanizmi, Türinq maşınında hesablama kimidir: qeyri-müəyyənlik halında (prosesində) hesablama prosesi dayanmır. Beləliklə, rekursiv funksiyalar içərisində tam təyin olunmaya yol verilir, yəni hissə- hissə funksiyalar g funksiyasının, hansı ki, μ- operatoru tətbiq olunur halında özü hissə- hissədir, $f(x_1, x_2, .., x_n) = \mu y(g(x_1, ..., x_n, y) = 0)$ funksiyası aşağıdakı şərt daxilində nəzərə almaqla hesablanır: Əgər $y = 0, 1, ..., i-1$ üçün $g(x_1, ..., x_n, y) \neq 0$ $g(x_1, ..., x_n, i)$ funksiyası təyin olunmayıb, onda həmçinin $f(x_1, ..., x_n)$ təyin olunmayıb. Məsələn, $f(x) = \mu y[y - x = 5]$ $x = 1$olduqda təyin olunmayıb (ancaq $y - 1 = 5$ tənliyinin həlli var, $y = 6$), necə ki, $y - 1 - 5$ funksiyası $y = 0$ olduqda təyin olunmayıb.

Hissə- hissə rekursiv funksiya o vaxt ümumirekursiv adlanır ki, əgər o hər yerdə təyin olunub.

Tutaq ki, $f(x_1, ..., x_n)$ funksiyası belə verilib: “hələlik $P(x_1, ..., x_n)$ doğrudur, $q_1(x_1, ..., x_n)$ hesablanır, bu halda $q_2(x_1, ..., x_n)$ hesablanır” $f(x_1, ..., x_n)$ funksiyası təkrar və ya tsikl adlanır ki, $q_1(x_1, ..., x_n)$, $q_2(x_1, ..., x_n)$- dən asılı və $P(x_1, ..., x_n)$ şərti ilə və $f(x_1, ..., x_n) = D(q_1, q_2, P)$ kimi işarə olunur.

Göstərmək olar ki, əgər q_1, q_2 və P primitiv-rekursivdirsə onda $f = D(q_1, q_2, P)$ hissə- hissə rekursivdir.

Bu nəticəni PR – B ayrılma operatorunun və S_m^n- yə, həmçinin Bom-Djakopanı teoreminə əsasən biz alarıq ki, istənilən proqram (kompüter üçün proqramlar mənasında) hissə- hissə primitiv – rekursiv kimi istifadə olunur.

İndi isə alqoritmlər üzərinə qoyulan əsas tələblərin hissə- hissə rekursiv funksiyalar üçün necə yerinə yetirilir.

Determinləşdirilməsi tam aydınlıqla təyin olunur sadə $0, x+1, U_m^n$ funksiyalarının hesablanmasında, həmçinin operatorların superpozisiyası, primitiv- rekursiya və μ- operatorunun əməllərin tam verilməsi ilə. Bütün uyğun addımlar birqiymətli yazılıb onların təyinində. Addımların oradaca elementarlığı və hesablamanın diskretliyi göstərilir.

Nəticə hissə- hissə rekursiv funksiyanın nəticəsidir, hansı ki, yazılmış operatorların həyata keçirilməsi prosesində hesablanır. (Oradaca elementarlıq). Əgər funksiya təyin olunmayıbsa, onda hesablama prosesi dayanmır.

İstənilən natural ədədin arqument şəklində seçilmə mümkünlüyü onun yəni hissə- hissə rekursiv funksiyanın kütləviyini təmin edir.

Hissə- hissə rekursiv funksiya anlayışı hesablanan funksiya anlayışının formalaşması olur (xüsusi halda Akkerman funksiyası ümumrekursivdir).

Bu ifadə Çörç tezisi kimi ifadə olunur:

Çörç tezisi: Hesablanan funksiyalar sinfi rekursiv funksiyalar sinifi ilə üst- üstə düşür, başqa sözlə hər bir hesablanan funksiya rekursiv funksiyadır.

(Çörç tezisi: Hər hansı alqoritmlə hesablanan bütün funksiyalar hissə-hissə rekursivdir)

Mövzu: **<u>Normal Markov alqoritmləri</u>**

<u>Plan</u>

1. Əlifba və sözlərin kodlaşdırılması
2. Sözlərin çevrilməsi
3. Normal alqoritmlər

1) Bildiyimiz kimi, alqoritmlərin xassələrindən biri də onların konstruktiv obyektlərlə işlənməsidir.Natural ədədlər, tam ədədlər, rasional ədədlər, onların cütləri, üçlükləri, tam və ya rasional əmsallı çoxhədlilər, rasional əmsallı xətti tənliklər və onların sistemləri belə obyektlərdir. Bu obyektləri müəyyən sonlu əlifbanın sözləri vasitəsi ilə təsvir etmək, başqa sözlə kodlaşdırmaq olar. qeyd edək ki, əlifba dedikdə işarələrdən (simvollardan) ibarət sonlu çoxluq başa düşülür. Təbiidir ki, həmin çoxluq boş deyil. Əlifbanı təşkil edən simvollar onun hərifləri adlanır. Qeyd edək ki, ixtiyari təbii dilin məsələn, azərbaycan, ingilis, türk və s. dillərinin əlifbasıda elə bu cür başa düşülür.

Əlifbanın hərflərindən ibarət hər bir sonlu ardıcıllıq bu əlifbanın sözü adlanır. Buradakı "söz" anlayışı təbii dildəki "söz" anlayışından xeyli fərqlidir; məsələn, lüöd azərbaycan dilində heç bir məna vermədiyindən bu dilin sözü deyil.

Ümumi halda isə, konkret məna daşımasa da biz əlifbanın simvollarının istənilən sonlu ardıcıllığından ibarət sözlərə baxacağıq.

Müəyyən obyektlərin kodlaşdırılmasına nümunələr göstərək.

Adi həyatda işlətdiyimiz kimi onluq say sistemindən istifadə etdikdə natural ədədləri $\{0,1,2,3,...,9\}$ əlifbasının sözləri şəklində, ikilik say sistemində isə $\{0,1\}$ əlifbasının sözləri şəklində yazmaq olar.

Yalnız bu simvoldan – cizgidən ibarət $\{1\}$ əlifbasını götürsək, bu əlifbada I,II,III, IIII,... sözlərinə uyğun olaraq 0,1,2,3,... ədədlərinin kodları kimi baxıla bilərş Tam ədədləri iki hərfdən ibarət olan $\{1,-\}$ əlifbasının sözləri vasitəsilə yazmaq olar.

Bu əlifba daha bir "/" (çəp xətt) hərfini əlavə etməklə alınan üçhərfli $\{1,-,1\}$ əlifbasının sözləri ilə rasional ədədləri kodlaşdırmaq olar. Bu əlifbada 1/3, -3/4, 2/1 ədədlərini uyğun olaraq, II/III, -III/IIII, III/II şəklində yazmaq olar. Tam ədədlərdən ibarət olan

$$\begin{pmatrix} 3 & -1 \\ -2 & 0 \end{pmatrix}$$

cədvəlini (matrisini) $\{0,1,-,*,\square\}$ əlifbasının sözləri ilə ("*" hərfi bir sətirdəki ədədləri "□" isə sətrləri bir-birindən ayırmaq üçündür)

$$\text{IIII}^{*}\text{-1}\square\text{-III}^{*}\text{0}$$

şəklində

$$\begin{cases} 2x-3y=4 \\ x+2y=-19 \end{cases}$$

tənliklər sistemini $\{0,1,2,...,9,+,-,=,*,x,y\}$ əlifbasında yazılmış

$$2x-3y=4^{*}x+2y=-19$$

sözü şəklində kodlaşdırmaq olar.

Alqoritmin ilkin verilənləri, aralıq və son nəticələri konstruktiv obyektlər olduğundan və bu obyektlər sonlu əlifbanın sözləri ilə təsvir edilə bildiyindən sözlər üzərində alqoritmlərə baxmaq fikri meydana çıxır.

Tutaq ki, A əlifbası cəmi iki hərfdən ibarətdir: $A=\{a,b\}$. A əlifbasının bütün sözləri çoxluğunu A^* ilə işarə edək. Onda *abb,ba,a,b,* $aaabb \in A^*$

Sözlə daxil olan hərflər sayına sözün uzunluğu deyilir. A əlifbasının A^* sözləri çoxluğuna daxil olan yuxarıdakı sözlərin uzunluğu uyğun olaraq 3,2,1,1,5- dir.

A^* çoxluğunda uzunluğu 2 olan dörd söz (aa, ab,ba,bb) , uzunluğu 3 olan səkkiz söz vardır. 1 uzunluqlu sözlərin sayı isə ikiyə bərzbərdir (a,b).

Heç bir hərf saxlamayan – “0” uzunluqlu sözün mövcud olmaması da qəbul edilir. Bunu boş söz (boş çoxluğun analoqu kimi) adlandırır və A şəklində yazırlar.

Daha bir mühüm anlayışa- bir sözün digərinə daxil olması münasibətinə baxaq.

Tutaq ki, müəyyən əlifbada L və M sözləri verilmişdir. Əgər L sözü M sözünün hissəsi olarsa, deyəcəyik ki, L sözü M sözünə daxildir (və ya L sözünün M sözünə girişi vardır). Bu zaman M sözü p_1Lp_2 şəklində göstərilir, burada p_1 və p_2 sözləri boş söz ola bilər. Bunların ikisi də boş söz olduqda L sözü elə M sözüdür və onlardan hər biri digərinə daxildir.

Ümumiyyətlə, L sözünün M sözünə bir neçə girişi ola bilər.

Misallar.

1) *L=ac; M=bacbba* olduqda L sözünün M sözünə (ikinci hərfdən başlayaraq) girişi vardır;

2) *babbbaaab* sözünə ab sözünün iki (birinci dəfə ikinci hərfdən, ikinci dəfə isə səkkizinci hərfdən) girişi vardır;

3) beb sözü abcbcbab sözünə iki dəfə (birinci dəfə ikinci hərfdən, ikinci dəfə isə dördüncü hərfdən daxildir).

Əgər M sözü p_1 p_2 şəklindədirsə və p_1 ən kiçik uzunluğa malikdirsə buna L- in M-ə birinci girişi deyəcəyik.

2)Tutaq ki, “$\leftrightarrow$” və “$\rightarrow$” hərflərini saxlamayan bir əlifbada L və M sözləri verilmişdir (L və ya M boş söz də ola bilər). Biz bir sözü digərinə çevirən

$$L \leftrightarrow M \text{ və ya } L \rightarrow M$$

əvəzləmələri şəklində verilən çevirmələrə baxacağıq. L $\leftrightarrow$ M şəklində verilmiş əvəzləməni R sözünə aşağıdakı qayda ilə tətbiq etmək olar. r sözünə L- in bir və ya bir neçə girişi varsa, onların hər biri M sözü ilə əvəz edilə bilər və əksinə, R sözünə M- in bir və ya bir neçə girişi varsa onların hər biri L sözü ilə əvəz edilə bilər.

Məsələn, a) $ab \leftrightarrow bcb$ əvəzləməsini *abcbcbab* sözünə dörd üsulla tətbiq etmək olar. Həmin sözə *bcb* nin iki girişinin (*abcbcbab*) ab ilə əvəz olunması *aabcbab abcabab* sözlərini *ab*- nin iki girişinin bcb ilə əvəz olunması isə *bcbcbcbab* və *abcbcbbcb* sözlərini verir.

b) $ab \leftrightarrow ba$ əvəzləməsi *bbb* sözünə tətbiq edilə bilməz. Çünki, bu sözə *ab* sözü, nə də *ba* sözü daxil deyildir.

İstənilən istiqamətlənmiş əvəzləmə adlanan L $\rightarrow$ M şəkilli əvəzləmənin R sözünə tətbiqi isə R-ə yalnız L sözünün girişi olduqda mümkündür. Bu zaman R-ə L-in hər bir girişi M ilə əvəz olunur, amma R-ə M-in girişi nəzərə alınmır.

Məsələn, a) $ab \rightarrow ba$ əvəzləməsi abaabba sözünə ancaq iki üsulla tətbiq edilə bilər, bunun nəticəsində uyğun olaraq, baaabba və abababa sözlərini alırıq.

Misal 1. $((10+2)\cdot(8-3)):(15-3)$ ifadəsinin ədədi qiymətinin tapılmasına $\{0,1,2,...,9,+,-,\cdot,:,\sigma\}$ əlifbasında yazılmış $((10+2)\cdot(8-3)):(15-3)$ sözünə aşağıdakı istiqamətlənmiş əvəzləmələrin tətbiqləri ardıcıllığı kimi baxmaq olar:

$$(10+2)\to 12,\ \ (8-3)\to 5,\ \ (15-3)\to 12,\ \ (12-5)\to 60,\ \ 60:12\to 5$$

$p\leftrightarrow A$ şəkilli əvəzləmə göstərir ki, əvvəla (həmin əvəzləməni “soldan sağa” tətbiq etdikdə) istənilən p- nin girişi əvəzinə “boş söz” yerləşdirmək olar, yəni verilmiş sözdən sadəcə olaraq, p-nin girişini atmaq olar (məsələn, bb$\leftrightarrow$A əvəzləməsini abbbbabb sözünə bu qayda ilə *bb*-nin ikinci girişinə tətbiq etsək ababbabb alarıq); ikincisi ($p\leftrightarrow A$ əvəzləməsini “sağdan sola” tətbiq etdikdə) istənilən sözün ixtiyari hərifinin arasında p sözünü yerləşdirmək olar (ixtiyari iki hərf arasında boş sözün olması təbiidir) Bundan başqa, p$\leftrightarrow$A əvəzləməsini “sağdan sola” tətbiq etdikdə ixtiyari sözün əvəzinə və həm də sonuna p sözünü yazmaq olar.

3) Normal alqoritmlər nəzəriyyəsi A.A.Markov tərəfindən keçən əsrin 40-cı illərinin sonunda işlənilmişdir. Bu alqoritm hər hansı əlifbada sözlərin düzəlməsi qaydasını araşdırır.

Əgər A və B – iki əlifbadırsa, belə ki, $A\subseteq B$ (A B- də yerləşir) onda B əlifbası A-nın genişlənməsi adlanır. Sözlərilatın hərflərinin böyükləri ilə işarə edəcəyik: P,Q,R,...; alqoritmləri $M_1, M_2,...,$ hərfləri ilə işarə edəcəyik.

A əlifbası üzərində alqoritm işi prosesi ardıcıl sözlərin yaranmasından təşkil olunur.Tutaq ki, M alqoritmi başlanğıc verilmiş P sözünü A əlifbasında Q sözünə çevirir. Bunu aşağıdakı kimi işarə edəcəyik:

$$M : P \Rightarrow Q \qquad (1)$$

ola bilsin ki, proses Q-nün yaranması ilə başa çatsın, hansı ki, biz onun verilmiş söz üzərində M alqoritminin işinin nəticəsi adlandıracayıq. Deyəcəyik ki, M alqoritminin P sözünə tətbiq olunmuşdur, əgər bu söz üzərində iş prosesi qurtarır, yəni bu verilmiş söz üzərində nəticə var.

Sonralar

$$!M(P) \quad (2)$$

yazılışın M alqoritmi P sözünə tətbiq olunmuşdur kimi işarə edəcəyik.

Tutaq ki, B- əlifba M_1 və M_2 - B üzərində alqoritmlərdir. M_1 və M_2- ni B-yə yə nəzərən ekvivalent adlandıracayıq, əgər ixtiyari P və Q sözü üçün (B əlifbasında) aşağıdakı iki şərt ödənilir.

1) əgər $M_1 : P \Rightarrow Q$, onda $M_2 : P \Rightarrow Q$

2) əgər $M_2 : P \Rightarrow Q$ onda $M_1 : P \Rightarrow Q$

Markov normal alqoritmləri adlanan anlayışa baxaq.

(P,Q) söz cütlüyündə Markov əvəzləməsi aşağıdakı əməliyyat adlanır.

Verilmiş R sözündə P sözünün birinci yerləşdiyini təyin edir (əgər varsa) və R-in qalan hissəsini dəyişmədən onu Q sözü ilə əvəz edirik. Alınan söz (P,Q) Markov əvəzləməsinin R sözünə tətbiqinin nəticəsi adlanır. Əgər R-də P yoxdursa, onda hesab edilir ki, (P,Q) əvəzləməsi R-ə tətbiq olunmur.

Xüsusi halda Markov əvəzləmələri boş sözlərlə əvəzləmə ilə olan əvəzləmələrdir: $(\wedge, Q)$, $(P, \wedge)$, $(\wedge, \wedge)$.

Misal 1. R=шрам sözündə (pa,ap) əvəzləməsini tətbiq edək. Bu halda P=pa, Q=ap, nəticə söz шрам olacaq.

Misal 2. Cədvəldə Markov əvəzləmələri misallarına baxılıb.

Dəyişdirilən söz	Markov əvəzləməsi	Nəticə
541551678	(41551,00)	500678
барабан	$(\delta a, \wedge)$	рабан
щрам	(ра,ар)	шарм
функция	$(\wedge, \beta -$	β - функция
книга	(кн, $\wedge$)	ига
мама	$(\wedge, \wedge)$	мама
дожд	(ga,т)	tətbiq olunmur

Burada $P \to Q$ yazılışı (P,Q) əvəzləməsi düsturu ilə olur. P- sol tərəf, Q- sağ tərəf adlanır. Birçox əvəzləmə son əvəzləmə adlanır. Bunu $P \to \bullet Q$ ilə işarə edirlər.

A əlifbasında nizamlı sonlu siyahı yerdəyişmə düsturu

$$S: \begin{cases} P_1 \to (\bullet) Q_1 \\ P_2 \to (\bullet) Q_2 \\ \dots\dots\dots\dots\dots \\ P_n \to (\bullet) Q_n \quad burada \quad n \geq 1 \end{cases}$$

A əlifbasında normal alqoritm sxemi adlanır. $(\bullet)$- onu göstərir ki, o bu yerdə dura bilər, ola bilsin iştirak etmir.

A əlifbasında normal Markov alqoritmi aşağıdakı kimi ilə qurulan $\{R_i\}$ ardıcıllığının qurulma qaydası adlanır, bu əlifbada verilmiş R sözü ilə. Başlanğıc R_0 sözü ardıcıllıqda R sözü götürülür. Tutaq ki, $\forall i \geq 0 \quad R_i$ sözü baxılan ardıcıllıqla qurulur hələlik sona çatmayıb.

$$R_0 \to R_1 \to R_2 \to \dots \to R_{m-1} \to R_m, \; R_0 = R, \; R_m = S$$

olduqda $\langle A, S \rangle$ cütü A əlifbasında normal alqoritm adlanır.

Deməli, A əlifbasında normal alqoritm A əlifbası və bu əlifbada normal sxem vasitəsilə verilir.

Mövzu: **<u>Markov alqoritmləri üzərində əməliyyatlar.</u>**

<u>Plan</u>

1. Alqoritmlərin ekvivalentliyi
2. Teoremlər.

Tutaq ki, M_1 və M_2 alqoritmlərdir. P- isə sözdür. Biz $M_1(P)=M_2(P)$ ifadəsini o faktlara tətbiq edəcəyik ki, ya M_1 və M_2 alqoritmləri P sözünə tətbiq olunmur ya da hər ikisinə tətbiq olunur və buna görə $M_1(P)=M_2(P)$. A əlifbası üzərində iki M_1və M_2 alqoritmi A-ya görə tamamilə ekvivalentdir adlandıracayıq ki, əgər A əlifbasında ixtiyari P sözü üçün $M_1(P)=M_2(P)$ bərabərliyi ödənilərsə o alqoritmləri A əlifbasına nəzərən ekvivalentdir adlandıracayıq ki, əgər $M_1(P)=M_2(P)$ hər dəfə ödənilir, harada ki, P A-da sözdür və heç olmazsa $M_1(P)$ və ya $M_2(P)$ sözlərindən biri təyin olunub və A-da sözdür.

Tutaq ki, k-$\{*,1\}$ əlifbasını göstərir, N-bütün natural ədədlər çoxluğu,φ- n arqumentli hissə-hissə effektiv hesablanan funksiyadır (yəni hər hansı N^n alt çoxluğunu N-ə inikas etdirir). Tutaq ki, A 1-i saxlayan əlifbanın, hər bir n natural ədədi üçün induksiya ilə $\overline{0}=1, \overline{n+1}=\overline{n}1$ təyin edək, beləliklə, $\overline{1}=11, \overline{2}=111$ və s. $\overline{n}$ sözü rəqəmlər adlanır. İndi $\{\wedge \to \bullet 1$ sxemi ilə normal M alqoritmi təyin edək.

A əlifbasında ixtiyari P sözü üçün $M(P)=1P$ alarıq. Xüsusi halda ixtiyari M natural ədədi üçün $M(\overline{n})=n+1$. Misal üçün $(\overline{2,3,1})$ $111*1111*11$ simvolu ilə işarə edilir. M_φ ilə k- da uyğun alqoritmi işarə edək, yəni belə alqoritmi $M_\varphi\left(\left(\overline{k_1,...,k_n}\right)\right)=\varphi(k_1,...,k_n)$, əgər $\varphi(k_1,...,k_n)$ -nin qiyməti təyin olunubsa və M_φ- əks halda təyin olunmayıbsa. Həmçinin M_φ alqoritmi $\overline{(k_1,...,k_n)}$ şəklində olan sözdən fərqli sözə tətbiq olunmur.

Əgər k üzərində normal M normal alqoritmi mövcuddursa və k- ya görə M_φ- yə tamamilə ekvivalentdirsə onda, φ funksiyasını Markova görə hissə- hissə hesablanan funksiya adlandıracayıq.

Əgər φ funksiyası hər yerdə təyin olunubsa,yəni ixtiyari n natural ədədlərdən seçilmiş və Markova görə hissə-hissə hesablanandır, onda onu biz Markova görə hesablanan adlandıracayıq.

Biz o vaxt normal alqoritm haqqında qapalıdır deyəcəyik ki. Əgər onun sxemi $\wedge \to \bullet Q$ şəkilli əvəzləməni saxlayır.

İki alqoritmi aşağıdakı kimi birləşdirmək olar. İxtiyari başlanğıc verilənlərə əvvəlcə birinci alqoritmi tətbiq edirik, sonra nəticəyə ikincini. Bu yazılış yeni alqoritm təşkil edir- iki alqoritmin kompozisiyasını.

İki normal alqoritmin kompozisiyası normal alqoritmdir. A əlifbasında M_3 normal alqoritm beledir; belə ki, $\forall P$ sözü üçün A-da, $M_3(P) = M_2(M_1(P))$. Bu normal alqoritm M_1 və M_2-nin kompozisiyası adlanır.

Tutaq ki, M- hər hansı normal alqoritmdir (A əlifbasında) B- A-nın hər hansı genişlənməsidir. Başlanğıc sxemdə M alqoritminə bütün mümkün olan əvəzləmə formulları $b \to b$ şəkilli, hara da ki, b B\A-da hərfdir, əlavə edək. Beləliklə, alınan sxem M_B normal alqoritmini təyin edir (B əlifbasında), hansı ki, B\A- dan olan heç bir sözə tətbiq olunmur və belə ki, $M_B(P) = M(P)$ $\forall P$ üçün A- dan olan M_B alqoritmi tamamilə A əlifbasına nəzərən M alqoritminə ekvivalentdir və M- in formal genişlənməsi adlanır B əlifbasına.

Tutaq ki, M_1 və M_2 normal alqoritmləri verilmişdir, uyğun olaraq A_1 və A_2 əlifbalarında. $A = A_1 \cup A_2$ əlifbasına baxaq və formal M_{1A} və M_{2A} genişlənmələrinə M_1 və M_2 alqoritmlərinin A əlifbasında M_3 kompozisiyası M_{1A} və M_{2A} alqoritmlərinin M_1 və M_2-nin normal kompozisiyası adlanır və M_2OM_1 simvolu ilə işarə olunur. Ümumi halda $M_n o...oM$,altında $M_n o(...0M_3 o(M_2 oM_1)...)$-i başa düşəcəyik. M_3 A üzərində normal alqoritmdir,belə ki, $M_3(P) = M_2(M_1(P))$ $\forall P$ sözü üçün A_1 əlifbasında və bundan başqa M_3 belə sözlərə P-yə tətbiq olunur. A əlifbasında hansı ki, bu şərtləri ödəyir:

1) P- A_1- də olan sözdür;

2) M_1 P-yə tətbiq olunur;

3) M_2 $M_1(P)$- yə tətbiq olunur.

Qəbul edək ki, B əlifbası A-nın genişlənməsidir və tutaq ki, P- B əlifbasında ixtiyari sözdür. P- nin A- da P^A proyeksiyası P- dən B\A-yə daxil olan hərfləri silməklə alınan sözdür. Qısa olaraq yazılmış sxem

$$\{\xi \to \wedge \ (\xi \in B \setminus A)$$

M_{B_1A}- proyektləşdirici normal alqoritm verib, yəni B-dən olan ixtiyari P sözü üçün $M_{B_1A}(P) = P^A$

<u>Teorem 1.</u> Tutaq ki, $M_1,...,M_k$ normal alqoritmlərdir və onların əlifbalarının birləşməsidir. Onda A üzərində normal M alqoritmi mövcuddur ($M_1,...,M_k$ alqoritmlərini birləşdirən) belə ki,

$$M(P) = M_1^{\neq}(P)^{\neq} M_2(P)...M_k^{\neq}(P)$$

ixtiyari P sözü üçün A-da, $M_i^{\neq}$- M_i- nin A- ya təbii genişlənməsidir.

Lemma 1) Tutaq ki, M A- da normal alqoritm və α ixtiyari hərf ($\alpha \neq A$) onda elə N normal alqoritmi var ki, $A \cup \{\alpha\}$ - da $\forall P \in A$ üçün

$$N(P) = \begin{cases} \alpha P & \text{əgər} \quad M(P) = \wedge \\ P & \text{əgər} \quad M(P) \neq \wedge \end{cases}$$

və N alqoritmi o sözlərə tətbiq olunur ki, M – tətbiq olunsun.

2) Onda elə $A \cup \{\alpha\}$- da elə M_3 normal alqoritmi var ki, $M_3(P) = M_1(P)$ və $M_3(\alpha, P) = H_2(P)$ $\forall P \in A$ üçün

Teorem 2. Tutaq ki, M_1, M_2, M_3 normal alqoritmlər, A- onların əlifbalarının birləşməsidir. Onda A-da elə H normal alqoritmi var ki,

$$M(P) = \begin{cases} M_2(P) & \text{əgər} \quad P - A - \text{nın sözüdür və} \quad M_3(P) = \wedge \\ M_1(P) & \text{əgər} \quad P - A - \text{nın sözüdür və} \quad M_3(P) \neq \wedge \end{cases}$$

A- nın o sözlərinə tətbiq olunur ki, hansı ki, M_3 tətbiq olunur.

Teorem 3. Tutaq ki, M_1 və M_3- normal alqoritmlər, A- onların əlifbalarının birləşməsidir. M_{11} və M_{33} uyğun olaraq M_1 və M_3-ün formal genişlənmələridir A- da. Onda A-da M_2 normal alqoritmi mövcuddur, hansı ki, M_{11}- in təkrarı, M_{33}-lə idarə olunan.

Markov (Çörç- Harkov) tezisi.

İntiutiv mənada hesablanan hər bir funksiya Markov mənada da hesablanandır (və ya hər bir alqoritmi normal alqoritm şəklində təsvir etmək olar).

Mövzu: **Qeyri- məhdud registrli maşınlar və orada hesablanan funksiyalar.**

Plan

1. Qeyri- məhdud registrli maşınlar (QMRM)
2. QMRM- də hesablanan funksiyalar.

Bİz burada, hansı ki, daha yüksək dərəcədə müasir kompüterlərə- QMRM-ə yaxın olan alqoritmik modeli şərh edəcəyik. Əsas təriflərə baxaq.

QMRM- aşağıdakılardan təşkil olunub:

1) lentdən, sonsuz sayda registri saxlayan, $R_1, R_2, R_3, \ldots$ ilə işarə olunan, hansı ki, hər bir istənilən an hər hansı natural ədədi saxlayır. R_n- də yerləşən ədədi biz r_n ilə işarə edəcəyik.

2) Sonlu sayda əmrlər siyahısından təşkil olunmuş P proqamından. Əmrlər aşağıdakı 4 şəkildə olur.

a) $Z(n)$ sıfırlaşdıran əmri- QMRM-i məcbur edir ki, R_n- in məzmununu “0”-la əvəz etsin; həmçinin $0 \rightarrow R_n$ və ya $r_n := 0$ ilə işarə olunur ($r_n := 0$ belə oxunur: r_n-ə 0 təqdim olunur)

b) $S(n)$ vahid əlavə edən əmri – QMRM-i məcbur edir ki, R_n-ə 1 əlavə olunsun; həmçinin $r_n + 1 \rightarrow R_n$ və ya $r_n := r_n + 1$ (r_n-ə $r_n + 1$ təqdim olunur)

c) $T(m, n)$ ünvan ötürən əmri – QMRM-i məcbur edir ki, R_n- də yerləşəni r_n ədədini R_m -də yerləşən r_m ədədi ilə əvəz etsin; $r_m \rightarrow R_n$ və ya $r_n := r_m$ (r_n-ə r_m təqdim olunur) ilə işarə edilir.

d) $J(m,n,q)$-şərti keçid əmri QMRM-i məcbur edir ki, orada yerləşən R_m və R_n registrlərini müqayisə etsin, sonra əgər $r_m = r_n$ olarsa, onda P proqramının q-cü əmrini yerinə yetirməyə keçir, əgər $r_m \neq r_n$ onda P- dəki şərti keçid əmrindən sonrakı əmri yerinə yetirməyə keçir. Əgər şərti keçid o şəkildə mümkün deyilsə, necə ki, P-də q-yə nəzərən əmr azdır, onda QMRM işi dayandırır.

Sıfırlaşdırıcı vahid əlavə etmə və yeni ünvan əmrləri hesabi adlanırlar. Hesablama aparmaq üçün QMRM $P = I_1, I_2, ..., I_S$ proqramları və başlanğıc konfiqurasiya ilə yüklənməlidir, yəni $R_1, R_2, R_3, ...$ registrlərindəki $a_1, a_2, a_3, ...$ natural ədələr ardıcıllığı ilə yüklənməlidir. QMRM I_1 əmrini yerinə yetirməklə işə başlayır. Tutaq ki, hesablamanın hər hansı anında QMRM I_k əmrini yerinə yetirir. Onda bundan sonra QMRM aşağıdakı kimi təyin olunan növbəti əmrlərin yerinə yetirilməsinə keçir.

* Əgər I_k hesabi əmrdirsə, onda növbəti əmrlə hesablanan I_{k+1} olacaq;

* Əgər $I_k = J(m,n,q)$ -sə, onda növbəti əmrlə hesablanan I_q olacaq, əgər $r_m = r_n$ və əks halda I_{k+1} (r_m və r_n R_m və R_n-də cari yerləşəndir uyğun olaraq).

Beləliklə, QMRM işini o vaxta kimi davam etdirir ki, hansı ki, o mümkündür. Hesablama yalnız və yalnız o vaxt dayanır ki, nə vaxt ki, növbəti əmr yoxdur, yəni QMRM indicə I_k əmrini yerinə yetirmişdi və növbəti hesablanacaq I_ϑ əmri var harada ki, $\upsilon > s$. Bu aşağıdakı hallarda yerinə yetirilə bilər:

a) Əgər $k = s$ (P-də axırıncı əmr yerinə yetirildi) və I_S hesabi əmrdir.

b) Əgər $I_k = J(m,n,q)$, $r_m = r_n$ və $q > s$

c) Əgər $I_k = J(m,n,q)$, $r_m \neq r_n$ və $k = s$.

Biz deyəcəyik ki, I_k əmrinin yerinə yetirilməsindən sonra hesablama dayandı; sonuncu konfiqurasiya- nəticə (yerləşdirmə) bu addımdakı registrdə (yerləşən) təşkil olunan $r_1, r_2, ..., r_n$ qiymətlər ardıcıllığıdır.

Misal 1. Aşağıdakı proqrama baxaq

$I_1\ Y(1,2,6)$

$I_2\ S(2)$

$I_3\ S(3)$

$I_4\ J(1,2,6)$

$I_5\ J(1,1,2)$

$I_6\ T(3,1)$

QMRM- də belə proqramla hesablamanı verək, başlanğıc konfiqurasiyada

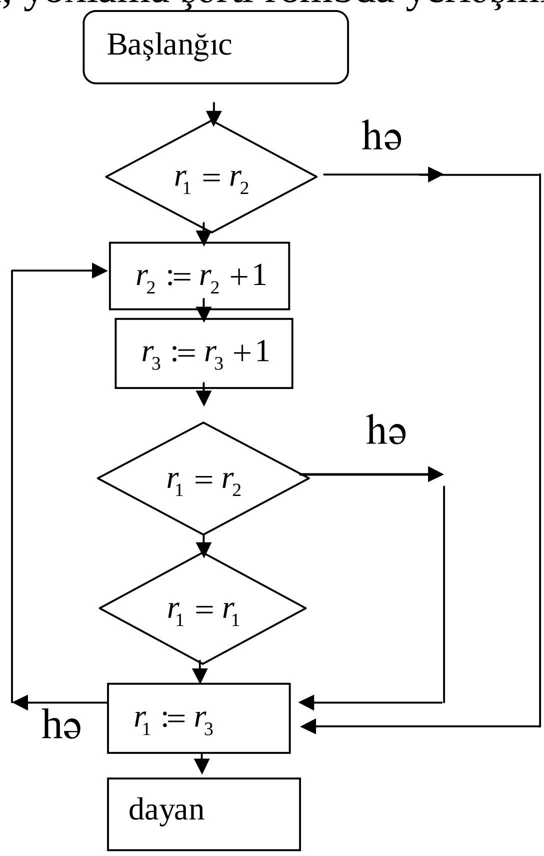
R_1	R_2	R_3	R_4	R_5	
9	7	0	0	0	...

QMRM- də iş prosesinə baxaq: verilmiş proqramla fikirləşmirik ki, faktiki olaraq hansı funksiyanı hesablayırırq. Hesablamanın gedişini belə vermək olar ki, maşının konfiqurasiyasını ardıcıl olaraq yuxarıdan aşağıya aşağıdakı əmrlərlə birlikdə, hansı ki, o verilmiş addımda keçir.

	R_1	R_2	R_3	R_4	R_5 ...	Növbəti əmr
Başlanğıc konfiqurasiya						
	9	7	0	0	0	I_1
	9	7	0	0	0	I_2 $(r_1 \neq r_2)$
	9	8	0	0	0	I_3
	9	8	1	0	0	I_4
	9	8	1	0	0	I_5 $(r_1 \neq r_2)$
	9	8	1	0	0	I_2 $(r_1 = r_2)$
	9	9	1	0	0	I_3
	9	9	2	0	0	I_4
	9	9	2	0	0	I_6 $(r_1 = r_2)$
Sonuncu konfiqurasiya	2	9	2	0	0	I_7 - dayan

Burada hesablama dayanır, harada ki, proqramda 7-ci əmr yoxdur.

Misal 2. Misal 1-dəki proqramın blok-sxeminə baxaq. (hesabi əmrlər düzbucaqlıda, yoxlama şərti rombda yerləşmişdir)



2) Tutaq ki, f N^n çoxluğundan götürülmüş funksiyadır. $(N^n \in N, n \geq 1)$ f funksiyası QMRM- də hesablanandır nə deməkdir? Təbii ki, cavab olaraq termində $f(a_1, \ldots, a_n)$ funksiyasının P proqramında qiymətinin hesablanması (başlanğıc $a_1, a_2, ..., a_n, a, 0, ...$ konfiqurasiyalı) yəni biz $P(a_1, a_2, ..., a_n)$ şəkilli hesablamaya baxırıq. Buna görə $f(a_1, a_2, ..., a_n) = b$, onda $P(a_1, a_2, ..., a_n)\downarrow$ ($\downarrow$- göstərir ki, hesablama əvvəl- axır dayanır) və hər hansı registrdə, məsələn R_1-də, b-ni tapmaq olar, yəni $r_1 = b$ sonuncu konfiqurasiyada. Əgər hər hansı $a_1, a_2, ..., a_n$ seçimində $f(a_1, a_2, ..., a_n)$ təyin olunmayıb, onda təbii ki, $P(a_1, a_2, ..., a_n)\uparrow$ ($\uparrow$- hesablama əvvəl-axır dayanmır). Beləliklə, biz belə bir tərifə gəlirik.

Tutaq ki, f N^n-də hissə- hissə funksiyadır. Tutaq ki, P – proqramdır, $a_1, a_2, ..., a_n, b \in N$.

$P(a_1, a_2, ..., a_n)$ hesablanması b-yə yığılır, əgər $P(a_1, a_2, ..., a_n)\downarrow$ və R_1 registrində sonuncu konfiqurasiyada b tapılır; bu belə yazılır $P(a_1, a_2, ..., a_n)\downarrow b$.

QMRM – P proqramlı, QMRM- f –i hesablayır, yalnız və yalnız o vaxt ki, $(a_1, a_2, ..., a_n) \in Dom(f)$.

f funksiyası QMRM-də hesablanandır, əgər elə QMRM varsa ki, QMRM f –i hesablayır.

QMRM-də hesablanan funksiyalar sinfi b_i ilə, n yerli QMRM hesablanan funksiya b_n ilə işarə olunur.

<u>Misal 3.</u> $f(x, y) = x + y$ $(x + y)$- i biz alırıq, x-ə 1-i (uyğun əmrlərdən istifadə etməklə) y dəfə toplayırıq $(x + y)$- in hesablanması başlanğıc x,y,0,0,... konfiqurasiya ilə; bizim proqram 1-ə r_1- i əlavə etməklə davam edir, R_2- dən sayğac kimi istifadə olunur. Hesablama prosesində tipik konfiqurasiya belədir:

R_1 R_2 R_3 R_4 R_5

x+k	y	k	0	0	...

Proqram dayan – a proyektləşdirilmişdir. Buna görə R_1 registrində x+y alınır, hansı ki, tələb olunurdu. Beləliklə bizim məsələni QMRM belə proqramla həll edir.

I_1 $Y(3,2,5)$ I_3 $S(3)$

I_2 $S(1)$ I_4 $J(1,1,1)$

Beləliklə, $f(x,y) = x + y$ QMRM-də hesablanan olur.

$Dom(f)$- f funksiyasının təyin oblastı

$Ran(f)$- f –in qiymətlər çoxluğudur.

Mövzu: <u>**Hesablama və həll olunmanın müxtəlifliyi. Alqoritm nəzəriyyəsinin ekvivalentliyi.**</u>

<u>Plan</u>

1. Müxtəlif alqoritm nəzəriyyəsinin ekvivalentliyi.
2. Alqoritmlərin nömrələnməsi.

Biz bir neçə nəzəriyyələrlə tanış olduq, hansı ki, hər biri alqoritm anlayışını dəqiqləşdirir. Sual olunur, bu nəzəriyyələr öz aralarında neçə əlaqəlidirlər? Buna aşağıdakı fundamental teorem cavab verir.

<u>Teorem 1.</u> Aşağıdakı funksiyalar sinfi (natural ədədlərdə verilmiş və natural qiymətlər alan) üst-üstə düşür.

a) Türinqə görə hesablanan bütün funksiyalar sinfi,

b) bütün hissə- hissə rekursiv funksiyalar sinfi

v) bütün normal hesablanan funksiyalar sinfi

q) Bütün QMRM-də hesablanan funksiyalar sinfi

Bu deyilənlər daha tezis deyil, riyazi teoremdir,riyazi teoremdir, hansı ciddi isbat oluna bilər (isbatı verməyəcəyik).

Baxılan bu teoremin mahiyyətini və əhəmiyyətini aydınlaşdıraq, hansı ki, Türinq maşınları nəzəriyyəsi, normal Markov alqoritmləri, rekursiv funksiyalar və QMRM eyni güclüdür. Müxtəlif dövlətlərin alimləri bir-birindən asılı olmayaraq alqoritmin intuitiv anlayışını öyrənərək və hesablanmanı, nəzəriyyələr yaratmış, verilmiş anlayışları gösytərən bu nəzəriyyələr eynigüclüdür. Əgər bu siniflərdən hər hansı biri digərindən geniş olduğu göstərildikdə onda uyğun Çörç, Markov və Türinq tezisi yalana çıxarılır. Məsələn, əgər normal hesablanan funksiyalar sinfi hissə- hissə rekursiv funksiyalar sinfindən genişdirsə, onda normal hesablanan ancaq hissə- hissə rekursiv olmayan funksiyalar vardır. Onun normal hesablanan olmasından o alqoritmik hesablanan olması, alqoritmin intiutiv başa düşülməsi və onun hissə-hissə rekursiv olmasını, Çörç tezisini yalana çıxarır. Ancaq qeyd etdiyimiz teorem doğrudur və belə funksiyalar yoxdur, hansı ki, yeni köklü həqiqətin Türinq, Markov və Çörç tezislərinin doğruluğunu alırıq. Qeyd edək ki, yenə və müxtəlif alqoritmlər nəzəriyyəsi mövcuddur və bütün onlar üçün həmçinin isbat olundu, onların eynigüclülüyü baxılan nəzəriyyədə xüsusi diqqət çəkən Türinq tezisidir. Türinq maşının tərifinin spesifikliyi gücü ilə o daha yüksək həqiqətdir.

Birinci prinsip, hansı ki,alqoritm anlayışını konkret dəqiqləşdirən, A.Çörç tərəfindən XX əsrin 30-cu illərində formalaşdırılmışdır.

Bir az sonra Türinq maşını anlayışının nəzəriyyəsi A.Türinq tərəfindən kəşf edilmiş və öz prinsipini yaratmışdır. “f funksiyası hesablanandır” ifadəsinin “f funksiyasını hesablayan alqoritm mövcuddur” ifadəsi ilə ekvivalentliyi bəzən hesablanan funksiya və alqoritm anlayışını qarışdırır. Bunlar arasında fərq- bu funksiya ilə onun verilmə üsuludur. Məsələn, primitiv- rekursiv funksiyalar ola bilər ki, çoxlu primitiv- rekursiv yazılışa malik olsun. Primitiv- rekursiv yazılışları həmçinin ekvivalentliyə görə siniflərə ayırmaq olar.

İndi isə dediyimiz nəzəriyyənin tətbiqi tərəfinə nəzər yetirək. Alqoritmlər nəzəriyyəsinin əsas nəticələri invariantlığı şəklində, onun tətbiqi əhəmiyyəti heç cür onun alqoritmlərin nəzəri modelində praktikada istifadə olunanı ilə əlaqəli deyil. Məsələn, Türinq maşını müasir kompüterlərdən çox aralıdır (fərqlidir), rekursiv funksiyalar isə proqramlaşdırma dillərindən fərqlidir.

2) Alqoritmlərin nömrələnməsi.

Alqoritmlər çoxluğu hesabidir. Konkret alqoritmik modelə müraciət etmədən izah etmək olar ki; istənilən alqoritm sonlu sayda yazılışla vermək olar, ancaq bütün konkret sözü qeyd olunmuş əlifbada hesabidir.

Alqoritmlər çoxluğunun hesabiliyi göstərir ki, alqoritm və natural sıranın ədədi arasında qarşılıqlı birqiymətli uyğunluq var, yəni $\varphi : N \to A^*$ tipli funksiya harda ki, A^* - alqoritmlər çoxluğudur.

Belə $\varphi(n) = A,\ n \in N,\ A \in A^*$ funksiyası alqoritmin nömrələnməsi adlanır, onun arqumenti n – A alqoritminin nömrəsidir. φ-ni nömrələyəndə. Tərs funksiyada mövcuddur.

$$\varphi^{-1}(A_n) = n$$

Mövzu: **Assosiativ hesab.**

Plan

1. Assosiativ hesab anlayışı
2. Misallar

Alqoritmin xassələrindən biri onların konstruktiv obyektlərlə əlaqədar olmasıdır. Natural, tam, rasional əmsallı çoxhədlilər, rasional əmsallı tənliklər və onların sistemləri konstruktiv obyektlərdir. Bu obyektləri müəyyən sonlu əlifbanın sözləri ilə təsvir etmək, başqa sözlə kodlaşdırmaq olar.

Alqoritmin ilkin verilənləri, aralıq və son nəticələri konstruktiv obyektlər olduğundan bu obyektlər sonlu əlifbanın sözləri ilə təsvir edildiyindən sözlər üzərində alqoritmlərə rast gəlirik.

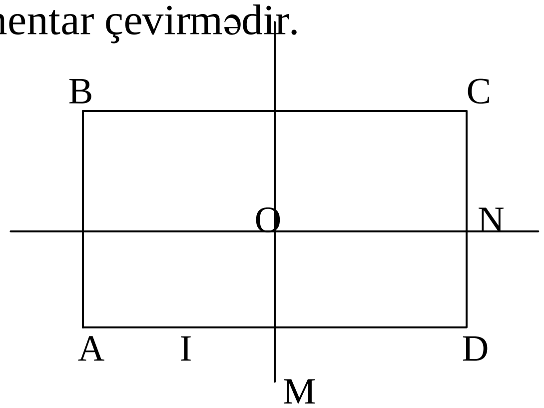
Müəyyən əlifbanın sözləri çoxluğuna və əvəzləmənin hər hansı sonlu sisteminə assosiativ hesab deyilir. Məsələn, gəmilərdə bəzi əmrlərin (komandaların) verilməsi üçün xüsusi kiçik bayraqlardan istifadə olunur. Bu zaman siqnallar bayraqların ancaq iki vəziyyətindən istifadə edir. Həmin vəziyyətləri a və b adlandıraq. Gizli olmaq üçün eyni əmr (komanda) müxtəlif şəkillərdə (müəyyən sayda) verilə bilər. Tutaq ki, vəziyyət onlara uyğun olan sözlərin birinin digərindən müəyyən sayda əməliyyatların köməyi ilə alındıqda müəyyən əmri (komandanı) bildirir. Məsələn, ba sözü ab sözü ilə əvəz olunur, ardıcıl gələn aa və bb hərfləri atılır. İstənilən yerə bu hərflər qrupu əlavə olunur və s.

Burada əmrlər (komandalar) $\{a,b\}$ əlifbasının sözlərinə uyğundur. Amma bu əlifba görünüşünə görə müxtəlif olan sözlər sonsuz saydadır. Bəzi müxtəlif sözlər eyni komandanı (əmri) bildirə bilər.

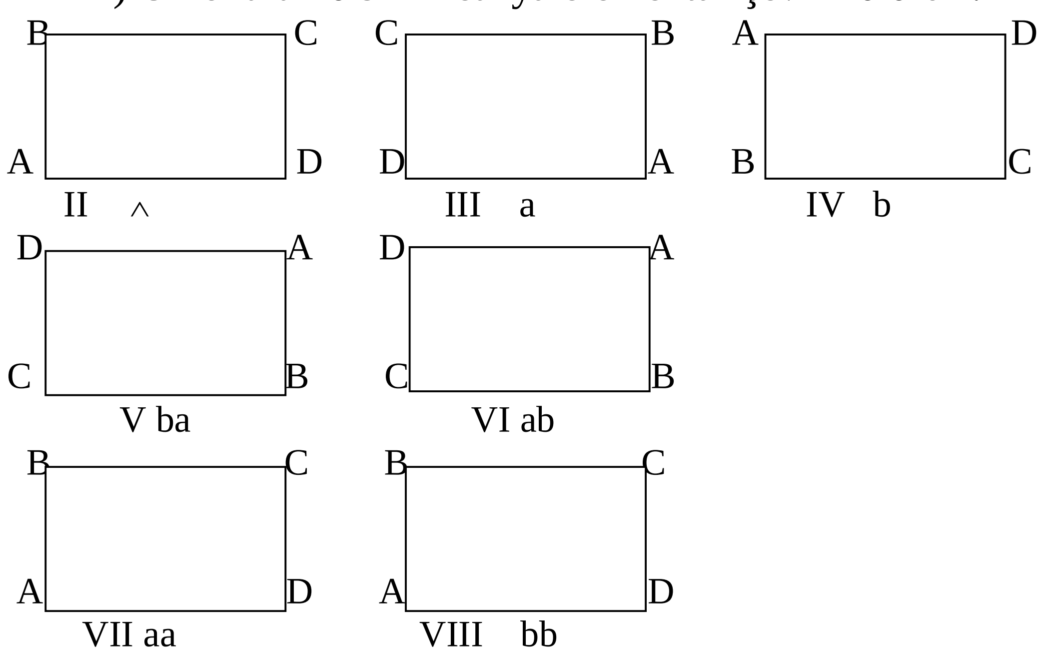
Eyni əmr (komanda) bildirən müxtəlif sözləri ekvivalent sözlər adlandırırıq. Ekvivalentlik işarəsi “~” ilə işarə edirik. Boş söz siqnalının heç bir əmr (komanda) vermədiyini göstərir.

P və Q sözünün biri digərinə əvəzləmənin bir dəfə tətbiq olunması ilə çevrilə bilirsə P və Q sözlərinə assosiativ hesabla qonşu sözlər deyilir.

ba; ab;abb;a;bab;abb qonşu sözlərdir. ABCD düzbucaqlısını simmetriya mərkəzindən (O mərkəzindən keçən) OM oxu üzrə simmetriya elementar çevirmədir.



2) On oxu üzrə simmetriya elementar çevirmələrdir.



Eyni sözləri bərabərliklə işarə edək.

$$ba = ab \qquad (1)$$

$$aa = \wedge \qquad (2)$$

$$bb = \wedge \qquad (3)$$

Bu hesabda düzbucaqlının öz-özünə çevrilməsi arasında əlaqə yaratmaq olar. (1), (2), (3) bərabərliklərindən çıxır ki, ixtiyari qonşu sözlər bərabərdir. Təklifin tərsi də doğrudur. $aa \to \wedge$, $bb \to \wedge$ əvəzləməsini tətbiq etməklə, aa...abb...b sözünün $\wedge\, a, b, ab$ sözlərinə ekvivalent olmasını alırıq.

aa və *bb* sözlərindən hər biri eynilik şevirməsidir. Buna görə də, onları boş sözlə təsvir edirik. Beləliklə,

$$ba = ab$$

$$aa = \wedge$$

$$bb = \wedge$$

bərabərliklərinin alarıq. (1)- (3) bərabərliklərini baxılan assosiativ hesabın əvəzləmələri ilə müqayisə etsək, həmin assosiativ hesabla düzbucaqlının özünə çevirmələri arasında əlaqə yaratmaq olar.

Doğrudan da (1)- (3) bərabərliklərindən çıxır ki, ixtiyari iki qonşu söz bərabərdir. Onda baxılan hesabda P və Q sözlərinin ekvivalentliyindən $P = Q$ alınır. Bu təklifin tərsi də doğrudur: yəni $P = Q$ olduqda $P \sim Q$ olur. Baxdığımız hesabın hər bir sözü $\wedge\, a, b, ab$ sözlərindən birinə ekvivalentdir. Doğrudan da, ixtiyari P sözünə $ba \leftrightarrow ab$ əvəzləməsini tətbiq etsək, həmin sözə ekvivalent olan və bütün a hərfləri bütün b hərflərdən solda gələn söz, yəni $\underbrace{aa...a}_{l}\underbrace{bb...b}_{m}$ sözünü alarıq; burada l və m sözdə, uyğun olaraq yerləşən a və b hərflərinin sayını göstərən ədəddir. Bundan sonra $aa \to \wedge$, $bb \to \wedge$ əvəzləməsini tətbiq etməklə, hər bir aa...abb...b sözünün $\wedge\, a, b, ab$ sözlərindən birinə ekvivalent olmasını alırıq.

$P = Q$ və $\wedge\, a, b, ab$ sözlərinin müxtəlif çevirmə təsvir etməsindən çıxır ki, P və Q sözləri göstərilən sözlərdən biri ilə ekvivalentdir. Deməli, $P \sim Q$.

İsbat etdiyimiz təklif həndəsi təsəvvürdən istifadə edərək, verilmiş assosiativ hesabda ixtiyari P və Q sözlərinin ekvivalentliyini müəyyən edə bilən aşağıdakı alqoritmi qurmağa imkan verir.

Alqoritm 1.

1. Düzbucaqlının P sözünə uyğun özünə çevrilməsini ardıcıl olaraq həyata keçirməli

2. Düzbucaqlının Q sözünə uyğun özünə çevrilməsini ardıcıl olaraq həyata keçirməli

3. Düzbucaqlının təpələrinin yerləşmə vəziyyətlərini müqayisə etməli. Onlar eyni olduqda 4-cü bəndə, əks halda 5-ci bəndə keçməli.

4. $P \sim Q$ prosesi başa çatdı

5. $P \nsim Q$ Proses başa çatdı

Bilirik ki, verilmiş hesabın ixtiyari sözü $\wedge\, a, b, ab$ sözlərindən birinə ekvivalentdir (onları etalon sözlər adlandırırlar).

Mövzu: **<u>Formal qrammatikanın əsasları.</u>**

<u>Plan</u>

1. Qramatikanın ümumi nəzəriyyəsində istiqamətlər

2. Qramatikanın növləri

Əsrlər boyu elmtəbii dilləri, dilin yaranma tarixini və onun qrammatikasını öyrənmişlər.

Müasir təbii dillərin səs, hərəkət və yazılı formaları vardır. Kiçik yaşdan başlayaraq uşaqları səs və yazılı dildən istifadə edərək tərbiyə edirlər: Lakin hərəkət tipli dillərdən də müəyyən lal- karlar istifadə edirlər. Digər canlılar da müxtəlif səslərlə, hərəkətlərlə ünsiyyətdə olurlar.

Alqoritmlər nəzəriyyəsində dilin yazılı formasından daha çox istifadə edirlər. Yazının meydana gəlməsi, xüsusi ilə kitab çapı təbii dillərin meydana gəlməsi, xüsusi ilə kitab çapı təbii dillərin dayanıqlığını təmin etdi. Təbii dildə hər bir cümlənin, hər bir sözün mənasına diqqət edilir. Elə sözlərdə var ki, onlar müstəqil heç bir məna daşımır, lakin cümlədə məna kəsb edir. Məsələn a əgər onda və s. Sözlərin ayrılıqda mənası yoxdur. Təbii dillərin sintaktikası və morfologiyası təyin edilir.

Sual olunur ki, təbii dilin hansı xüsusiyyəti alqoritmlər nəzəriyyəsində onun tətbiq imkanlarını azaldır. Bu xüsusiyyət dilin sintaksisinin semantikasından asılılığıdır, cümlələrin müxtəlif məna daşımasındadır, elə cümlələr düzəltmək mümkündür ki, onun forması ilə məzmunu arasında ziddiyyət yaradır.

EHM üçün yüksək səviyyıli dillərdə isə dilin sintaksisi onun semantikasından asılı deyil, elə sözlər düzəltmək olar ki, EHM onu “başa düşsün”, yəni həmin söz müəyyən məsələnin həllində rol oynasın, lakin təbii dildə heç bir məna daşımasın. Təbii dildə bir sözün bir neçə məna daşıması bizə məlumdur. Alqoritmlər nəzəriyyəsində isə yeganəlik tələb olunur.

Beləliklə, EHM- lər üçün yüksək səviyyəli dillərin yaradılması ilə əlaqədar olaraq təbii dildəki nöqsanlardan azad olan bir dilin yaradılmasına zərurət əmələ gəlmişdir.

Son illərdə müxtəlif dillərin və qramatikanın ümumi nəzəriyyəsi sahəsində 4 müstəqil məsələ qoyulmuş və onların həlli ilə əlaqədar 4 elmi istiqamət təyin olunmuşdur.

Birinci istiqamət formal və ya riyazi linqivistikanın yaradılmasıdır. Xüsusilə maşın tərcüməsi meydana çıxdıqda bu istiqamət sürətlə inkişaf etməyə başlamışdır.

Bu istiqamətlə yanaşı tamamilə müstəqil olaraq formal dinamik modellərin qurulması inkişaf etməyə başladı. Məsələn, “sonlu avtomatlar” modeli bu sahəyə misal göstərilə bilər. Bu anlayış ümumi maşınlar nəzəriyyəsi (Türinq maşınları) və daha sonra alqoritm nəzəriyyəsinə gəlib çıxır.

Buradan aralıq dinamik sistemlərin qurulması istiqamətində işlərin aparılmasını tələb etdi.

Bu istiqamətlərin 4-cüsü müasir riyaziyyatın ən böyük bölməsi olan alqoritmlərin ümumi nəzəriyyəsinin inkişafıdır. Burada “Normal Markov alqoritmləri”, “Ümumi rekursiv funksiyalar” və “Türinq maşınlarının ekvivalentliyinə” dair Çörç tezisi verildi.

Hesablama texnikasının inkişafı alqoritmlərin riyazi nəzəriyyəsində yeni məsələ qoydu alqoritmləri hesablamanın mürəkkəbliyinə görə təsnifata ayırmaq lazımdır. “Alqoritm” anlayışı ilə “Türinq maşını” anlayışların ekvivalenliyi sonlu avtomatika Türinq maşını modelləri arasında aralıq modellərin olmasının axtarışını tələb etdi. Buradan görünür ki, sayılan 4 istiqamət bir-biri ilə sıx əlaqədardır. Bütün bunlar isə alqoritmlərlə və modellərin dinamik sistemi ilə əlaqədar dillər nəzəriyyəsinin nəzəri əsaslarını işləməyi tələb edir. Burada təbii və süni dillərin quruluşunun öyrənilməsi qarşıya çıxır.

Formal qramatikanın öyrənilməsində tətbiq olunan riyazi aparat alqoritmlər və avtomatlar nəzəriyyəsinə tətbiq olunan aparatlara çox yaxındır.

Qramatika dedikdə müəyyən sonlu simvolların zəncirvari çoxluğu ilə verilən qaydaları başa düşürük. Bu zənciri çoxluğu söz formaları, söz birləşmələri və cümlələrin birləşmələri kimi şərh etmək olar.

Söz forması və ya sadəcə söz- morfem (morf) həmin ardıcıllığıdır. Morfem sözün çox kiçik qramatik nəzərə alınan hissəsidir.

Dilin sintaksisi –dildə cümlələrin qurulmasıdır və ya dilin qurulmuşudur.

Dilin semantikası- sintaksis qaydaların mənalarına görə araşdırılmasıdır.

Qramatikanın formalaşdırılmasının ilk və əsas tələbi onun hər bir cümləsinin struktur təsvirinin təyin edilməsidir. Bu qaydalar cümlənin hansı elementlərdən qurulduğunu hansı ardıcıllıqla yerləşdirilməsini təyin edir.

Qramatikanın ikinci tələbi- onun sonlu olmasıdır, yəni burada qaydalar çoxluğu sonludur.

Formal qramatikanın mücərrədləşdirmə ilə daha çox əlaqəsi vardır. Burada simvolları, sözbirləşmələrinə və cümlələrə çevrilir. Əgər adi qramatika cümlələri qurmaq üçün qaydalar çoxluğunun verilməsinə imkan yaradırsa, formal qramatika- qaydaların öyrənilməsini və təsvir edilməsinin müəyyən üsullarını öyrənməyi tələb edir.

Tanıyan, yaranan (doğuran) və çevrilən formal qramatikalar var.

Tanıyan- istənilən baxılan zəncirin düzgün olub- olmamasını yoxlaya bilir, müsbət cavab alındıqda bu zəncirin qurulmasına dair göstəriş verilir.

Yaranan (doğuran)- doğru zənciri quru bilir, onun qurulmasına göstəriş verir və heç bir düzgün olmayan zəncir qurmur.

Çevrici- istənilən düzgün qurulmuş zəncirin başqa bir düzgün inikası ala bilir və bu inikasın xarakteristikalarını verə bilir.

Doğuran qramatikaların siniflərinə baxaq. Doğuran qramatika nizamlanmış

$$G = (V_T, V_H, P, S)$$

sistemindən ibarətdir.

Burada V_T - sonlu boş olmayan simvollar olub G lüğətin əsas terminidir.

Terminal lüğıt- simvollar külliyatıdır ki, qramatikanı yaradır və ya dilin əsas sözlərindən ibarətdir ki, bu sözlərdən cümlələr düzəldilir, terminal simvollar daha kiçik simvollara bölünməyən simvollardır.

V_H - sonlu (boş olmayan) simvollardan ibarətdir. G-lüğətinin terminallarından (köməkçi) ibarətdir.

Qeyri terminal lüğət sinfin ilkin elementlərindən və ya sintaksis tipindən ibarətdir. V_T və V_H-in elementləri uyğun olaraq terminal və qeyri- terminallardan ibarətdir.

$V = V_T \cup V_H$ isə G qramatikasının sonlu simvolları olub, onun söz lüğəti adlanır.

V- nin elementlərindən təşkil olunan sonlu ω ardıcıllığın V- də zəncir adlanır. Boş zəncir $\wedge$ kimi işarə olunur. Bu sonlu ardıcıllığın hədlərinin sayı zəncirin uzunluğu adlanır. V qramatikası sözlərinin zənciri birləşmə simvolu ilə alınır $(\cap)$.

Doğuran qramatika alqoritm deyildir.

Mövzu: **Avtomatlar nəzəriyyəsinin elementləri.**

Plan

1. Məntiqi elementləri, məntiqi qurğular və avtomatların strukturu
2. Avtomatların sintezi üçün alqoritmlər.

Avtomatlar nəzəriyyəsi nəzəri kibernetikanın əsas bölmələrindən biridir. Burada avtomat adlanan riyazi modellər öyrənilir. Bu modellər vasitəsi ilə real mövcud olan və ya reallaşdırıla bilən diskret zaman ərzində diskret informasiyaları (texniki, bioloji və s.) emal edilən qurğulardır. Avtomat anlayışı rəqəmi hesablayıcı texnikanın və idarəetmə maşınların, həmçinin alqoritmlər nəzəriyyəsinin və riyazi məntiqin tələbindən yaranmışdır. Burada sonlu, sonsuz, ehtimallı, avtonom işləyən, determinalı və s. avtomatlar ola bilər. Maksimum dərəcədə ən ümumi avtomatların yaradılması məsələsinin tam həll olunduğunu hələ demək olmaz, ancaq idarə etmək sistemlərin xüsusi halına baxmaq olar. Avtomat terminin ilk rüşeymi 1930-cu illərdə alqoritmlər və releyli qurğular nəzəriyyəsində yaranmışdır. Lakin 1950-ci ildən sonra bir problem kimi meydana çıxmışdır. Məlumdur ki, alqoritmlər nəzəriyyısində sonlu avtomatlar kimi Türinq maşınları geniş öyrənilmişdir və göstərilmişdir. İxtiyari informasiyaları effektiv çevirmək üçün hər dəfə ixtisaslaşdırılmış avtomatlar yaratmaq lazım deyildir, bir universal avtomat yaratmaq kifayətdir.

Məhz universal hesablama texnikasının yaradılması bu məsələnin həllidir.

Avtomat nəzəriyyəsində cəbrin,riyazi məntiqin, kombinator analizin (qraf nəzəriyyəsi daxil olmaqla) və ehtimal metodlarından istifadə edilir, ona görə avtomatların cəbri nəzəriyyəsi, mücərrəd avtomatlar nəzəriyyəsi və s. nəzəriyyələr vardır.

Avtomatlar hərflər ardıcıllığını hərflər ardıcıllığına çevricilərdir. Bu qurğular nə maddə, nə də enerji emal etmir, siqnallarla işləyir. Müəyyən siqnallar vasitəsi ilə daxil olan “sözü” başqa dilin sözünə çevirir. Bu termin texnikada işlənən “avtomat” sözündən tamamilə fərqlənir.

Məntiqi əməlləri reallaşdıran- konyuktor, dizyunktor və inventor qurğuları vardır. Əsas bu funksiyaların elektrik realizasiyası üçün batareya, aşarlar və lampa kifayətdir. Lampanın yanması baxılan məntiqi əməlin düzgün yerinə yetirildiyini bildirir.

Və ya məntiqi əməli (dizyunksiya) iki açardan heç olmazsa biri qapandıqda lampa yanır. Belə funksiyanı yerinə yetirən qurğu dizyunktiv adlanır.

Və məntiqi əməli (konyuksiya) – hər iki açar qapalı olduqda və ancaq onda lampa yanır. Belə qurğu konyuktor adlanır

Deyil (inkar) məntiqi əməli açar qapalı olmadıqda lampa yanır. Bu növ qurğu invetor adlanır.

Sadə məntiq əməllərinin nəticələri siqnal yaradan qurğular məntiqi element adlanır. Məntiqi elementlər ən sadə avtomatlardır.

Konyuktur, dizyunktur və invetorlardan ibarət olan sxemlər funksioanl sxemlər adlandırılır.

Avtomatın daxili quruluşu onun strukturu adlanır. Avtomatların strukturunu düstur və ya sxemlər vasitəsi ilə təsvir etmək olar. Əgər avtomatın struktur düsturu məlumdursa, burada məntiqi elementlərin sayı və elementlərin birləşdirilməsi sxemini təyin etmək olar.

Avtomatların yaradılması üçün bir qayda olaraq aşağıdakı alqoritmlərdən istifadə olunur.

1. Avtomatın işləməsini təmin edən şərtləri təyin etmək.

2. Layihələşdirilən avtomatı “sehrli qutu” səviyyəsində təsəvvür etməli, bu avtomatın girişləri və çıxışları sayını müəyyənləşdirməyə imkan verir.

3. Layihələşdirilən avtomatıın işini təsvir edən cədvəl qurmalı.

4. Cədvəldən istifadə edərək avtomatın Bul funksiyasını yazmalı.

5. Struktur düsturu minimallaşdırmalı

6. Sonuncu struktur düstura görə avtomatın funksional sxemini tərtib etməli.

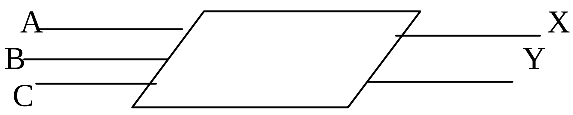
Məsələn, bir rəqəmdən ibarət üç ikilik rəqəmləri cəmləyən avtomatın layihələndirilməsi məsələsinə baxaq. Məsələni başa düşmək üçün analoji bir məsələyə baxaq. Zavodun A,B,C sxemləri var. Bu sxemləri X və Y generatoru olan elektrik stansiyası təmin edir. X generatorunun gücü Y-in gücündən 2 dəfə artıqdır. Sxemlərdən yalnız birinin işlənməsi üçün Y generatorunu, 2 sex üçün X generatorunu işə salmaq kifayətdir.

Növbəti A,B və C sexlərindən stansiyaya daxil olan siqnalları izləyir və lazım olan generatoru işə salır. Növbətçini əvəz edən avtomat düzəltmək tələb olunur.

Layihəçilər məsələnin yuxarıdakı şəkildə verilməsini avtomatın sözlərinə təsviri adlandırırlar. Layihəçi tapşırıqla tanış olduqdan sonra onu “sehrli qutu” kimi təhlil edir.

1. Avtomat A,B,C sexlərindən siqnal alır, deməli onun üç girişi olmalıdır.

2. Avtomatın yaratdığı siqnallar 2 ünvana X və Y generatorlarına göndərilməlidir, yəni avtomatın 2 çıxışı olmalıdır.



Avtomatın işini təsvir edən cədvəli çəkək. 2-ci sətir göstərir ki, enerji yalnız C sexinə lazımdır. 8-sətir isə hər 3 sexə enerji verilməsini tələb edir.

Alqoritmin strukturuna bir qutuya qoyulmuş 2 qurğu kimi baxmaq olar. Onlardan birinin çıxışı X, digərinin çıxışı Y-dir.

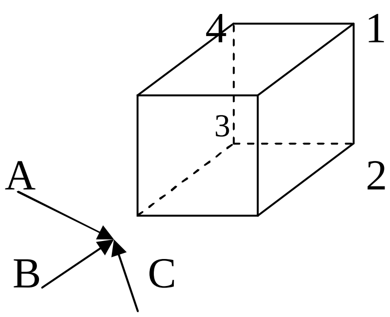
Beləliklə, çıxışda ya A,B və C sexlərinin üçündən ya B və C, ya A və B, ya da A və C ibarət olmaqla 4 siqnal alınır.

	A	B	C	X	Y
1	0	0	0	0	0
2	0	0	1	0	1
3	0	1	0	0	1
4	0	1	1	1	0
5	1	0	0	0	1
6	1	0	1	1	0
7	1	1	0	0	1
8	1	1	1	1	1

$$X = ABC + AB\overline{C} + A\overline{B}C + \overline{A}BC \qquad (1)$$

$$Y = ABC + A\overline{B}\overline{C} + \overline{A}\overline{B}C \qquad (2)$$

Sonrakı addımda düsturların minimallaşdırılması məsələsinə baxmalıyıq. Bu düsturda 3 mülahizə iştirak edir, ona görə kubdan istifadə edilə bilər.



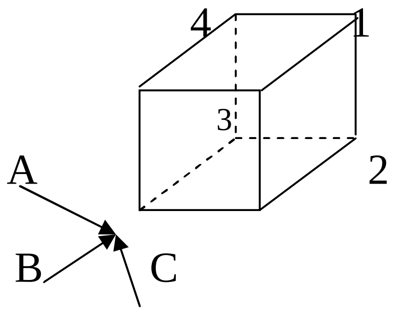
(1) düsturunun hədlərini kubun uyğun təpələrində təsvir edək. Əgər minimallaşdırma üçün həndəsi metoddan istifadə etsək X generatorunu idarə edən avtomat üçün yeni düstur alırıq:

$$X = AB + AC + BC$$

İndi kubun uyğun tillərində (2) düsturunu təsvir edək.

Bu halda nişanlanmış təpələrdən istənilən 2-si bir til üzərinə düşmədiyində (2) düsturunu minimallaşdırmaq mümkün deyil. Y generatorunun isə idarə edən avtomatın struktur düsturu əvvəlki kimi qalır.

İndi tutaq ki,A,B,C birmərtəbəli ikilik ədədlərdir. Onların cəmini tapmaq istiyirik. .



$$+\begin{array}{c} A \\ B \\ C \\ \hline XY \end{array}$$

Burada aşağıdakı hallar mümkündür.

1) $+\begin{array}{c} 1 \\ 1 \\ 1 \\ \hline 11 \end{array}$ 2) $+\begin{array}{c} 1 \\ 1 \\ 0 \\ \hline 10 \end{array}$ 3) $+\begin{array}{c} 1 \\ 0 \\ 1 \\ \hline 10 \end{array}$ 4) $+\begin{array}{c} 1 \\ 0 \\ 0 \\ \hline 01 \end{array}$ 5) $+\begin{array}{c} 0 \\ 1 \\ 1 \\ \hline 10 \end{array}$ 6) $+\begin{array}{c} 0 \\ 1 \\ 0 \\ \hline 01 \end{array}$ 7) $+\begin{array}{c} 0 \\ 0 \\ 1 \\ \hline 01 \end{array}$ 8) $+\begin{array}{c} 0 \\ 0 \\ 0 \\ \hline 00 \end{array}$

Görürük ki, avtomat “növbətçi” bir mərtəbəli 3 ikilik ədədi toplaya bilər. Deməli, ikilik say sistemində hesablamaları avtomatlara tapşırmaq olar.

Mövzu: **<u>Hesablama və həll olunmanın müxtəlifliyi.</u>**

Riyaziyyatda "hə" və "yox" cavabı tələb edən sonsuz sayda məsələlər vardır. Elə bir metod vardır ki, onun köməyi ilə eyni bir sinfin bütün suallarına sonlu addımlardan sonra cavab vermək mümkün olsun?

Eyni tipli məsələlərdən ibarətt sinfin hər bir məsələsinə "hə" və "yox" cavabını verən ümumi metodun olub-olmaması problemi həll problemi adlanır. Əgər sinfin hər bir məsələsinə "hə" və "yox" cavabını verməyə imkan yaradan ümumi bir metod mövcud olsa həmin sinifə həll olunan sinif, metodun özü isə həll prosedurası- həllalqoritmi və ya sadəcə alqoritm adlanır.

Məsələn: 1) a natural ədədi b natural ədədinə bölünürmü? Bu məsələ eyni tip məsələlərin sonsuz sinifini əhatə edir, çünki natural ədədlərdən düzəldilmiş (a,b) cütləri sonsuz saydadır. a və b –ni konkret natural ədədlər götürməklə xüsusi məsələlər düzəldirik.

2) $a_0x^n + a_1x^{n-1} + ... + a_{n-1}x + a_n = 0$

tam əmsallı cəbri tənliyinin rasional kökləri varmı?

Məktəb riyaziyyatında bu suala cavab verilir. Əgər bu tənliyin $\frac{p}{q}$ rasional kökü varsa p ədədi a_n- nin, q isə a_n- in böləni olmalıdır, a_n-nin və a_0-ın bütün bölənlər üçün yoxlama aparmalıyıq. Deməli, tənliyin rasional kökləri olub-olmamasını yoxlamaq üçün ümumi alqoritm var.

3) $ax + bx + c = 0$ tənliyinin a,b və c tam ədədlər olduqda tam həllərini tapmaq üçün həll prosedurası vardır.

Həll problemi ümumi halda məntiq elmində izah edilir. Burada mülahizələrdən və prodikat anlayışından istifadə edilir.

Haqqında “doğrudur” və ya “yalandır” hökmlərinin yalnız biri ödənilən hər bir nəqli cümləyə mülahizə deyilir. Prodikat isə bir və ya bir neçə dəyişəndən asılı olan təkliflərdir ki, bu təkliflər dəyişənlərin konkret qiymətlərində doğru və ya yalan olduğunu söyləmək mümkündür. Dəyişəndən asılı olan bir, iki, üc və s. yerli prodikatlar vardır (bir, iki, və s. funksiyalar). Həll problemində formula anlayışından istifadə edilir. Formula hər hansı təklifi özündə birləşdirən simvolik yazıdır, cəbri ifadədir. Məsələn $V = \frac{1}{3}SH$ piramidanın həcmini göstərən formuladır. Latınca formula forma anlayışının bir növ dəyişdirilmiş şəklidir.

Formula o zaman eynilik kimi doğru hesab edilir ki, onun tərkibinə daxil olan dəyişənlərin bütün qiymətlərində alınan mülahizələr doğrudur. Formula daxil dəyişənlərin hər hansı qiymətlərində doğru olsa informasiya icra olunmalıdır deyirlər. Əgər formulaya daxil olan dəyişənlərin bütün qiymətlərində yalan qiymət alırsa formula icra olunmayan adlanır.

İndi də qarşıya çıxan məsələyə diqqət yetirək: Elə yeganə prosedura (alqoritm) göstərin ki, onunla hər bir formulun doğru və ya yalan olmasını göstərmək mümkün olsun. Bu alqoritm varsa onun köməyi ilə formulun icra olunub olunmaması məsələsini də həll edə bilərik.

Qoyulmuş bu məsələ həll problemi adlanır. Bu problem təkcə mülahizələr üçün deyil, istənilən məntiqi sistem üçün qoyulur. Qeyd edək ki, hər hansı formal üçün üç məsələ qoyulur: “Formal ifadə formuldurmu?”, “Verilmiş sonlu formullar ardıcıllığı isbatdırmı?”, “Verilmiş formula isbat olunandırmı?”.

Həll problemi ilə bağlı olan məsələ sahəsində çoxlu riyaziyyatcılar fəaliyyət göstərmiş və davam etməkdədir, bu problem alqoritmlər nəzəriyyəsi ilə bağlıdır.

Həll problemi həmçinin hesablama problemi ilə bağlıdır və ona görə hesablama probleminə aid olan bəzi məsələləri öyrənmək.

Hesablama problemi Riyaziyyatda hesablama anlayışına tez- tez rast gəlirik. Burada eyni tipli xüsusi məsələlərin siniflərinə baxılır ki, bu siniflərin məsələsi üçün sonlu sayda addımdan sonra cavab göstərməyə imkan verən ümumi metodun aolub-olmaması problemi həmin sinif üçün hesablama problemi adlanır, sinfə hesablanan məsələlər sinfi, axtarılan metodlara hesablama metodu və ya hesablama proseduraı və ya hesablama alqoritmi deyirlər.

Tutaq ki, $f(x_1, x_2, ..., x_n)$ funksiyasının hesablanması məsələsi qoyulur. Burada nəyi başa düşürük? Əgər arqumentlərin verilmiş konkret qiymətləri üçün funksiyaların qiymətini tapmağa imkan verən hər bir prosedura (qayda) varsa deyirik ki, funksiya hesablanandır.

Məsələn, $a, b \in N$ olduqda ƏBOB(a,b) nəyə bərabərdir? Bu məsələ natural ədədlər cütü üçün ümumi məsələdir.

Bu sinif hesablanandır, onun ixtiyari xüsusi məsələsi Evklid alqoritmi ilə hesablana bilər.

ƏBOB(18,24)= ƏBOB (18,6)= ƏBOB (12,6)= ƏBOB(b,b)=b

Bu alqoritmlər verilmiş istənilən bir cüt natural ədədin ƏBOB-u hesablanır.

$a \neq 0$, b və c ixtiyari həqiqi ədələr olduqda $ax^2 + bx + c = 0$ tənliyini həll etməli məsələlər sinfi hesablanandır. Hesablam alqoritmi mövcuddur.

$$x_{1,2} = \left(-b \pm \sqrt{D}\right)/2a, \quad D = b^2 - 4ac$$

xətti cəbri tənliklər sinfinin həlli üçün müxtəlif hesablama qaydaları vardır.

Həndəsədən pərgar və xətkeş vasitəsi ilə həll olunan məsələlər vardır. Çox qədimdən dairənin kvadraturası, bucağın triyeksiyası, kubun 2 qat böyüdülməsi məsələləri nümunə göstərilə bilər.

Pərgar və xətkeş vasitəsi ilə elə kvadrat qurmaq olarmı ki, verilmiş dairə bir böyüklükdə olsun. Burada məsələ üçün təyin olunan dairə ixtiyari ola bilər, ona görə burada sinif məsələlər vardır. Pərgar və xətkeş vasitəsilə ixtiyari bucağı üç bərabər yerə bölmək üçün ümumi üsul varmı?

Verilmiş tilinə görə elə kub qurmaq olarmi ki, onun həcmi verilmiş kubun tilindən 2 dəfə böyük olsun?

Pərgar və xətkeş vasitəsi ilə bu məsələlərin həllinin mümkün olması isbat edilmişdir, yəni elə alqoritm yoxdur ki, bu məsələlri həll etmək mümkün olsun?

Beləliklə, hesablama və həll problemi ədədi nəzəri funksiyaların hesablanmasına gətirilir. Ona görə də hansı ədədi nəzəri funksiyaların hesablanması üçün prosedураların (alqoritmlərin) olması məsələsi həll olunandır, başqa sözlə “hesablanan” funksiyalar sinifləri hansılardır? Sualına cavab axtarılmalıdır. Çoxlu sayda hesablanan funksiyaların olmasını bilirik.

Riyaziyyatçıları maraqlandıran problemlərdən biri olmayan funksiyaların varlığını göstərmək üçün prosedurаları tapmaqdır.

Cəbrin yaranması və inkişafı ilə əlaqədar olaraq məsələlərin həlli üçün alqoritmlər yaranmışdır.

Məşhur yunan alimi Diofant tənliklər sisteminin həlli üçün metodlar göstərmişdir. O “hesab” kitabında 130-a qədər müəyyən tənliklərin ayrı- ayrı üsullarla həllini vermişdir. Diofantdan sonra hind qiyaziyyatçıları daha ümumi metodlar tapmışlar.

VI əsrdə yaşamış hind riyaziyyatçısı Ariabxatta bir məchullu tənliyə gətirilən məsələni həll etmişdir.

VII əsrdə hind riyaziyyatçısı Bramequpta birməchullu birdərəcəli kvadrat tənliyə və tənliklər sistemınə baxmışdır. XI əsrdə ərəb riyaziyyatçısı Omar adlı himd riyaziyyatcısı kvadrat tənliyin həlli üçün alqoritm vermişdir. Daha sonra italyan alimləri Ferro, Tartal, Kardano, Ferrari, Bombelli artıq 3 və 4 dərəcəli tənliklərin alqoritmlərini vermişdir.

Cəbri məsələlərin həlli üçün hesablama alqoritmlərin kəşfini Viyet, Dekart vermiş onlar hərfi simvolikanın daxil edərək universal alqoritmlərə baxmışlar.

Hələ qədim dövrlərdə arximed tərəfindən sahələrin, həcmlərin hesablanmasına gətirilən hesablama məsələlərini həll etmişdir.

XVII əsrdə diferensial və inteqral anlayışları praktik hesablama metodları yaranmışdır.

Törəmənin, inteqralın hesablanması üçün hesablanan alqoritmləri yaranmışdır.

Beləliklə alqoritm anlayışı, hesablama metodları ilə bağlanaraq müasir kibernetika və hesablanan maşınlarında çox mürəkkəb məsələlər həllinə yol açmışdır. Maşın alqoritmləri yaranmış və hesablamaların çox böyük sürətlə aparılması məsələləri həll edilmişdir.

Primitiv- rekursiv operatorlar.

Plan

1.Primitiv-rekursiv hesablanan funksiyalar sinfinin qurulması

2. Rekursiya operatoru

3. Minumumlaşdrma operatoru.

Hesabi funksiyaların bütöv bir sahəsini əhatə edən və rekursiv hesablanan funksiyalar adlanan bu sinfi konstuktiv olaraq aşağıdakı qayda ilə qurulur. Əvvəlcə rekursiv hesablanan funksiyaların öyrəndiyimiz 3(üc) əsas ən sadə hissəyə ayrılmasını qeyd etmək lazımdır. Bunlar "eyniliklə sıfır", "bilavasitə sonra gələn" və "proyeksiyalama" üsullarıdır. Sonra isə rekursiv funksiyalar nəzəriyyəsində operatorlar adlandırılan elə üç qayda vardır ki, onların köməyi ilə yeni hesabi funksiyalar almaq mümkündür. Bu qaydalar aşağıdakılardır:

1. Əvəzləmə (superpozisiya) operatoru. Bu operatorun qurulması üsulu ilə biz tanış olmuşuq. $f(y_1,...,y_n)$ və $f_1(x_1,...,x_n)$ funksiyaları verildikdə onlardan $y_1 = f_1 \ (1 \le i \le m)$ əvəzləməsi nəticəsində alınan

$$F(x_1,...,x_n) = f(t_1(x_1,...,x_n)),..., f_m(x_1,...,x_n)$$

funksiyanın qurulması.

2. <u>Rekursiya operatoru.</u> Bu operator işi, biri $(n-1)$ sayda $x_1, x_2,...,x_{n-1}$ dəyişəndən asılı f_1, digəri isə göstərilən dəyişənlərdən başqa daha iki dəyişəni özündə saxlayan (yəni n+1 dəyişəndən asılı) f kimi iki funksiyaya görə n dəyişənli yeni funksiyanın qurulması prosesindən ibarətdir.

Əlavə iştirak edən iki arqumentdən f_1 funksiyasının arqumentləri ilə birlikdə yeni funksiyaya dəyişən kimi daxil olur, digəri isə rekursiya operatorunun yerinə yetirilməsində köməkçi rol oynayır.

Bunları uyğun olaraq y ilə işarə edək. x – i baş arqument, y-i isə köməkçi arqument adlandırmağı şərtləşək. rekursiya operatorunun R-lə işarə edək. Verilmiş f_1 və f_2 funksiyalarından R operatorunun köməyi ilə alınan funksiya f olsun.

f_1 və f_2 funksiyalarından R operatorunun tətbiqi nəticəsində f funksiyasının alınması prosesini $f = R[f_1, f_2(x, y)]$ şəklində göstərək.

Bir çox hallarda rekursiya operatorunu qurarkən $f(x_1, x_2,.., x_{n-1})$ funksiyasının asılı olduğu $x_1, x_2,.., x_{n-1}$ arqumenləri həm onun özünə, həm də qalan iki f_2 və f funksiyalarını yazarkən göstərməmək və yalnız onları nəzərdə saxlamaq daha əlverişli olur.

Belə halda rekursiya operatoru f_1 və f_2 funksiyaları daxil olan və aşağıdakı iki şərtlə müəyyən olunan funksiya şəklində təyin etmək olar.

$$f(0) = f_1, \quad f(x+1) = f_2(x, f(x))$$

Başqa sözlə, təyin edəcəyimiz f funksiyasını baş arqumentin sıfır qiymətində verilmiş f_1 funksiyasının qiymətinə bərabər, baş qrqumentin hər sonrakı qiyməti üçün isə verilmiş f_2 funksiyasının baş arqumentin ondan əvvəlki qiyməti və köməkçi arqumenti təyin edəcəyimiz f funksiyasının əvvəlki qiyməti ilə üst- üstə düşən qiymətlə əvəz etmək nəticəsində aldığı qiymətə bərabər hesab edəcəyik.

3. Minumumlaşdrma operatoru. Bu operator $n+1$ sayda arqumentdən asılı verilmiş f_1 funksiyasına görə n sayda arqumentdən asılı yeni f

funksiyasını qurur. Yox edilən arqument köməkçi hesab olunur və yenidən operatorun yerinə yetirilməsi üçün istifadə olunur. Bu operatoru μ hərfi və onun tətbiqini $f=\mu(f_1(x))$ kimi işarə olunur. Burada *x* yox edilən arqumentdir.

μ operatorunun iş prinsipi aşağıdakı kimidir. Köməkçi arqumentə sıfırdan başlayaraq o vaxta qədər ardıcıl qiymətlər vermək lazımdır ki, f_1 funksiyası sıfra bərabər qiymət olsun, f_1 funksiyası birinci dəfə belə bir sifir qiyməti alan kimi prosesi dayandırmalı və təsvir etdiyimiz prosesin həyata keçirildiyi anda əsas arqumentlərin qiymətlər yığımı üçün təyin edəcəyimiz *f* funksiayasının qiyməti olaraq köməkçi arqumentin f_1 funksiyasını sıfra cevirən həmin qiymətini götürmək lazımdır.

Başqa sözlə, fərz edək ki, x dəyişəninin heç olmazsa bir mənfi olmayan tam qiyməti və $x_1,...,x_n$ dəyişənlərin hər hansı qiymətlər yığımı üçün $f(x_1,...,x_n,x)=0$ olur. *x*- in belə qiymətlərindən ən kiçiyini $\mu x(f_1(x_1,x_2,...,x_n,x))=0$ kimi işarə olunur. Bu halda $f(x_1,...,x_n)=f_2(x_1,...,x_n,x)=0$ olunduğunu qəbul etmək olar. belə olduqda deyəcəyik ki, *f* funksiyası f_1 funksiyasından μ operatoru vasitəsilə köməkçi arqumentin f_1 funksiyasını sıfra çevirən ən kiçik qiymətinə görə qururlur. Bu səbəbdən də həmin operatora birinci sıfra görə qurulan operator və ya ən kiçik ədəd operatoru da deyirlər.

Qeyd edək ki, bazis funksiyalar və habelə əvəzləmə və rekursiya operatorlarının köməyilə alınan funksiyalar arqumentlərin bütün mənfi olmayan tam qiymətlərində təyin edildiyi halda, birinci sıfra görə qurulan operatorun köməyi ilə alınan funksiyalar arqumentlərin bəzi və hətta sonsuz sayda qiymətləri üçün təyin edilməyə bilər.