През този семестър ще процедираме аналогично на предходния семестър. Ще имаме две контролни – задачи и теория. Критериите ще публикувам в Мудъл. Единственото ново условие е, че крайната оценка е $\frac{2}{5}$ от задачи + $\frac{3}{5}$ от теория. Причината е съотношението упражнение : лекции.
През този семестър ще използваме за учебник книгата на Harry R. Lewis и Christos H. Papadimitriou Elements of the Theory of Computation. Причината да избера книгата е, че в нея освен доброто изложение на необходимите теми има и систематично изложение на оценка на сложността на алгоритмите, което е много важно за един добър програмист.
Както разбирате основното, с което ще се занимаваме е теория на изчислимостта, т.е. какво е алгоритъм, алгоритмично изчислима функция, какво може и какво не може да бъде изчислено. За да си отговорим на тези въпроси или да се убедим че нямаме задоволителен отговор до момента ще трябва да имаме добра и солидна основа, както теоретична така и практическа.
В курса през този семестър ще се занимаваме с различни изчислителни модели, като започнем от най-простите и стигнем до най-сложните.
В началото ще започнем с така наречените регулярни езици и крайни автомати. Това най-простите изчислителни модели с които ще започнем и както ще видим те са еквивалентни като клас езици, които определят. Регулярните езици имат природата да бъдат породени (генерирани). Крайните автомати пък играят ролята на разпознаватели.
След това ще разгледаме контекстно-свободните езици и стековите автомати. Това е следващия по сложност модел на изчисление, като контекстно-свободните граматики играят рол на генератори, а стековите автомати на разпознаватели и отново те са еквивалентни. Най-накрая ще се спрем на машините на Тюринг, които играят рол както на разпознаватели, така и на генератори. Ще се спрем на различни модификации и видове на машини на Тюринг, които се докосват до толкова актуалните квантови компютри.

Уводни дефиниции

Д. Нека A е множество и $R \subseteq A^2$ е релация в A. Рефлексивно и транзитивно затваряне R^* на R се определя индуктивно както следва:

а) Ако $(a,b) \in R$, то $(a,b) \in R^*$;

б) Ако $a \in A$, то $(a,a) \in R^*$;

в) Ако $(a,b) \in R^*$ и $(b,c) \in R^*$, то $(a,c) \in R^*$.

Д. Нека $F: A^n \to A$ и $B \subseteq A$. Затварянето B^* на B относно функцията F се определя индуктивно както следва:

а) Ако $a \in B$, то $a \in B^*$;

б) Ако $a_1, \dots, a_n \in B^*$ и $F(a_1, \dots, a_n) = b$, то $b \in B^*$.

Предния семестър ние разглеждахме един тип затваряне на множество от двоични функции. То беше частен случай на горната дефиниция. На всеки избор $f, g_1, \dots, g_n$, $f: J_2^n \to J_2$, $g_i: J_2^k \to J_2$, $i = 1, \dots, n$ ние съпоставяхме функция $h: J_2^k \to J_2$, която беше суперпозиция на $f, g_1, \dots, g_n$. С други думи $F: \mathcal{F}_2^{n+1} \to \mathcal{F}_2$, F функция (операцията суперпозиция), такава, че $F(f, g_1, \dots, g_n) = h$. И тъй като F играе роля на суперпозиция, ние говорехме за затваряне на множ. от двоични функции относно операцията суперпозиция.

Горната дефиниция може да се обобщи за много функции (операции).

Д. Нека A е множество, $F_1, \dots, F_k$ са функции, $F_i: A^{n_i} \to A$, $i = 1, \dots, k$, а $B \subseteq A$. Затварянето B^* на B относно операциите $F_1, \dots, F_k$ се определя индуктивно както следва:

а) Ако $a \in B$, то $a \in B^*$;

б) Ако $a_1, \dots, a_{n_i} \in B^*$ и $F_i(a_1, \dots, a_{n_i}) = b$, то $b \in B^*$.

Тази дефиниция също може да се обобщи.

Д. Нека $R \subseteq A^{n+1}$, A – множество, а $B \subseteq A$. Затварянето B^* на B относно релацията R се определя както следва:

а) Ако $a \in B$, то $a \in B^*$;

б) Ако $a_1, \dots, a_n \in B^*$ и $(a_1, \dots, a_n, b) \in R$, то $b \in B^*$.

Пример 1. Нека $A = \mathbb{N}$ - мн. на ест. числа и $F(x) = x+1$. Да разгледаме $B = \{0\}$. Тогава затварянето B^* на B отн. F е цялото $\mathbb{N}$, защото най-напред $0 \in B^*$. След то $1 \in B^*$ и т.н. Всички ест. числа са от B^*

Пример 2. Нека $A = \mathbb{R}$ - мн. на реалните числа и $F(x) = x+1$. Тогава ако пак $B = \{0\}$, отново $B^* = \mathbb{N}$.

Пример 3. Нека A е произволно множество и $R \subseteq A^2$ е релация на еквивалентност в A, а $B \subseteq A$. Тогава B^* (затв. на B^* отн. R) е обединението на всички класове на еквивалентност, които съдържат ел. от B.

Да обясня защо казах по-горе, че посочената дефиниция е обобщение на втората (а също и на третата). Ако имаме една релация $R \subseteq A^{n+1}$, то R може да се разглежда като така наречената „многозначна" функция, т.е. функция която има повече от една функционална стойност в общия случай. Например за фиксирано $R \subseteq A^{n+1}$ и фиксирана n-орка $(a_1, \dots, a_n)$ под мн-от функ. стойности на R можем да разбираме мн. $\{b \mid (a_1, \dots, a_n, b) \in R\}$ и можем да го означаваме с $R(a_1, \dots, a_n)$. Така всяка графика на функция е релация която изразява функция и в някои разглеждания се счита за функция.

Можем да разглеждаме опр. и много релации.

Д. Нека A е множ., $R_i \subseteq A^{r_i+1}$, $i = 1, \dots, n$ и $B \subseteq A$. Затварянето B^* на B отн. $R_1, \dots, R_n$ се опр. индуктивно така:

а) Ако $a \in B$, то $a \in B^*$;

б) Ако $a_1, \dots, a_{r_i} \in B^*$ и $(a_1, \dots, a_{r_i}, b) \in R_i$, то $b \in B^*$.

Азбуки, думи, езици. Операции върху думи и езици

Д. Азбука наричаме всяко крайно множество от символи. Примери за азбуки са българската, латинската, обединение на части от българската с части от английската или всяко множество от букви, цифри и символи.

Д. Нека Σ е азбука. Дума в азбуката Σ се нарича всяко изображение $w:\{1,\dots,n\}\to\Sigma$. Вместо $w(1),\dots,w(n)$ ние ще записваме още $w_1,\dots,w_n$, а самата дума ще пишем $w=w_1w_2\dots w_n$, т.е. редица от букви в Σ, както е стандартното разбиране на дума. Дължина на думата w се означава с $|w|$ и $|w|=n$, т.е. броя на буквите в редицата $w_1\dots w_n$. Има една специална дума, която се нарича празната дума и тя се означава с ε. $\varepsilon:\phi\to\Sigma$, т.е. д-ната на ε е 0, т.е. $|\varepsilon|=0$. Така $w:I_n\to\Sigma$ е дума, като при $n=0$, $I_n=0$ и $|w|=0$, т.е. $\varepsilon=w$.

Операции м/у думи: 1) Конкатенация на думи. Нека u и v са думи в азбуката Σ. Тогава конкатенацията на u и v се означава с $u\circ v$ и ако $w=u\circ v$, то w е дума с д-на $|u|+|v|$ и $w:I_{|u|+|v|}\to\Sigma$ като $w(i)=u(i)$, за $1\le i\le|u|$ и $w(j)=v(j-|u|)$ за $|u|+1\le j\le|u|+|v|$, т.е. ако $u=u_1\dots u_n$ и $v=v_1\dots v_m$, то $w=u_1\dots u_nv_1\dots v_m$. Нека отбележим без да го доказваме, че операцията конкатенация е асоциативна, т.е. $(u\circ v)\circ w=u\circ(v\circ w)$, но конкатенацията не е комутативна в общия случай.

Д. Казваме, че думата u е начало на думата v, ако съществува дума x, такава, че $u\circ x=v$. Казваме, че u е край на думата v, ако същ. x, такава, че $x\circ u=v$. И най-накрая казваме, че u е поддума на v, ако съществуват думи x,y такива, че $x\circ u\circ y=v$.

Оттук нататък за простота конкатенацията на думите u и v ще означаваме просто uv.

2) Степенуване на думи. Нека w е дума, а n е естествено число. Ще определим индуктивно w^n както следва:

а) $w^0 = \varepsilon$; б) $w^{n+1} = w^n \circ w$. Иначе казано $w^n = w \circ w \circ \ldots \circ w$ (n пъти).

3) Обратна дума (reversal) на дадена дума w. Означаваме я с w^R и отново дефиницията е индуктивна.

а) $|w| = 0$, то $w = \varepsilon$ и $w^R = \varepsilon$; б) $|w| = n+1$, $w = u \circ a$, $a \in \Sigma$, то $w^R = a \circ u^R$. Едно свойство на обратните думи, свързано с конкатенацията е $(u \circ v)^R = v^R \circ u^R$. Ще го докажете на упр.

Със Σ^* ще означаваме всички думи в азбуката Σ.

Д. Нека Σ е азбука. Език в азбуката Σ се нарича всяко множество от думи в азбуката Σ, т.е. всеки език L е подмножество на Σ^*.

Пример. Езици са $\emptyset$, Σ, Σ^*.

Д. Казваме, че буквата a, $a \in \Sigma$ се среща в думата w, ако a е поддума на w.

Често езиците се задават с някое свойство. Например:

$L_1 = \{w \mid w \in \Sigma^*$ и a се среща в $w\}$ и
$L_2 = \{w \mid w \in \Sigma^*$ и a участвува поне n пъти в $w\}$

Естествено a участвува n пъти в w, ако съществуват думи $u_0, u_1, \ldots, u_n$, такива, че $w = u_0 a u_1 a u_2 \ldots u_{n-1} a u_n$.

Да отбележим, че ако в една азбука Σ има поне 1 буква a, то Σ^* е изброимо, защото $\varepsilon, a, a^1, a^2, \ldots$ са елементи на Σ^*, които са различни.

Тъй като всеки език е множество, то м/у езиците могат да се прилагат теоретико-множествените операции като $\cup$, $\cap$, $\setminus$. Освен тези операции м/у езиците можем да прилагаме и специфични операции

1) Конкатенация на езици: Нека L_1 и L_2 са езици. Конкатенацията на L_1 и L_2 се означава с $L_1 \circ L_2$ и $L_1 \circ L_2 = \{w \mid w = w_1 \circ w_2$ за някои $w_1 \in L_1$ и $w_2 \in L_2\}$.

Друга операция м/у езици е степенуване

2) Степенуване: Нека L е език и n е естествено число. Тогава L^n се определя индуктивно:

а) $L^0 = \{\varepsilon\}$; б) $L^{n+1} = L^n \circ L$;

Да отбележим, че $\emptyset^0 = \{\varepsilon\}$.

3) Звезда на Клини за език: Нека L е произволен език. L^* може да се определи по два начина:

3а) $L^* = \bigcup_{n=0}^{+\infty} L^n$; 3б) $L^* = \{w \mid w \in \Sigma^*$ и $w = w_1 \circ \dots \circ w_k$ за $w_1, \dots, w_k \in L$ и $k \in \mathbb{N}\}$. Това включва и $k=0$

Така $\emptyset^* = \{\varepsilon\}$.

Използва се още и L^+, като $L^+ = \bigcup_{n=1}^{+\infty} L^n$. .

Обърнете внимание, че $\varepsilon \in L^*$ винаги, но $\varepsilon \in L^+$ само ако $\varepsilon \in L$. Освен това означението Σ^* се съгласува със звездата на Клини на Σ.

Регулярни езици

Както казахме език е произволно множество от думи. Отбелязахме още, че всяка азбука с контретна буква съдържа изброимо много думи. Следователно в една непразна азбука (каквито ние обичайно ще разглеждаме) множеството на езиците са неизброимо много. От конструктивна, т.е. от алгоритмична гледна точка е добре да можем да разпознаваме езиците ако не всичките, то тези които използваме с алгоритмични цели. Ясно е, че няма кое да разпознаваме (от алг. гледна точка) всичките. Затова ще обърнем внимание на една съвкупност от езици, които се пораждат от краен брой символи. За да разберем какво точно значи това да разгледаме следния пример: Нека L е следния език $L = \{w \mid w \in \{0,1\}^*$ и w съдържа едно или 2 срещания на $1\}$

Това можем да го опишем по следния начин, използвайки $\{0\}, \{1\}$ и операциите $\cup, \circ$ и *.

$$\{0\}^* \circ \{1\} \circ \{0\}^* \circ ((\{1\} \circ \{0\}^*) \cup \emptyset^*)$$

Може да се провери, че $L =$ езикът, зададен в предния ред.

Така ние стигаме до регулярен израз.

Д. Нека Σ е азбука. Регулярен израз над (в) Σ се определя индуктивно както следва:

а) $\emptyset$ и всяка буква от Σ е регулярен израз;

б) Ако α и β са регулярни изрази, то $(\alpha\beta)$, $(\alpha \cup \beta)$, α^* са регулярни изрази.

С помощта на регулярен израз α ние ще определим регулярен език $\mathcal{L}[\alpha]$, определен от регулярния израз α.

Д. Индуктивна дефиниция на $\mathcal{L}[\alpha]$ с индукция относно построението на α: а) Ако $\alpha = \phi$, то $\mathcal{L}[\alpha] = \phi$;

б) Ако $\alpha = a \in \Sigma$, то $\mathcal{L}[\alpha] = \{a\}$;

в) Ако $\alpha = (\alpha_1 \alpha_2)$, то $\mathcal{L}[\alpha] = \mathcal{L}[\alpha_1] \circ \mathcal{L}[\alpha_2]$;

г) Ако $\alpha = (\alpha_1 \cup \alpha_2)$, то $\mathcal{L}[\alpha] = \mathcal{L}[\alpha_1] \cup \mathcal{L}[\alpha_2]$;

д) Ако $\alpha = \alpha_1^*$, то $\mathcal{L}[\alpha] = (\mathcal{L}[\alpha_1])^*$

Д. Казваме, че един език L е регулярен, ако същ. рег. израз α, такъв, че $\mathcal{L}[\alpha] = L$.

С други думи, регулярните езици се генерират (пораждат) от регулярните изрази. Затова се казва, че рег. изрази са генератори на регулярните езици.

Да разгледаме един пример. Нека $\alpha = ((a \cup b)^* a)$. Тогава да "пресметнем" $\mathcal{L}[\alpha]$.

$\mathcal{L}[\alpha] = \mathcal{L}[(a \cup b)^*] \quad \mathcal{L}[a] = (\mathcal{L}[(a \cup b)])^* \circ \mathcal{L}[a] = (\mathcal{L}[a] \cup \mathcal{L}[b])^* \circ \{a\} =$
$= (\underbrace{\{a\} \cup \{b\}}_{\{a,b\}})^* \circ \{a\} = \{w \mid w \in \{a,b\}^* \text{ и } w \text{ завършва с буквата } a\}$

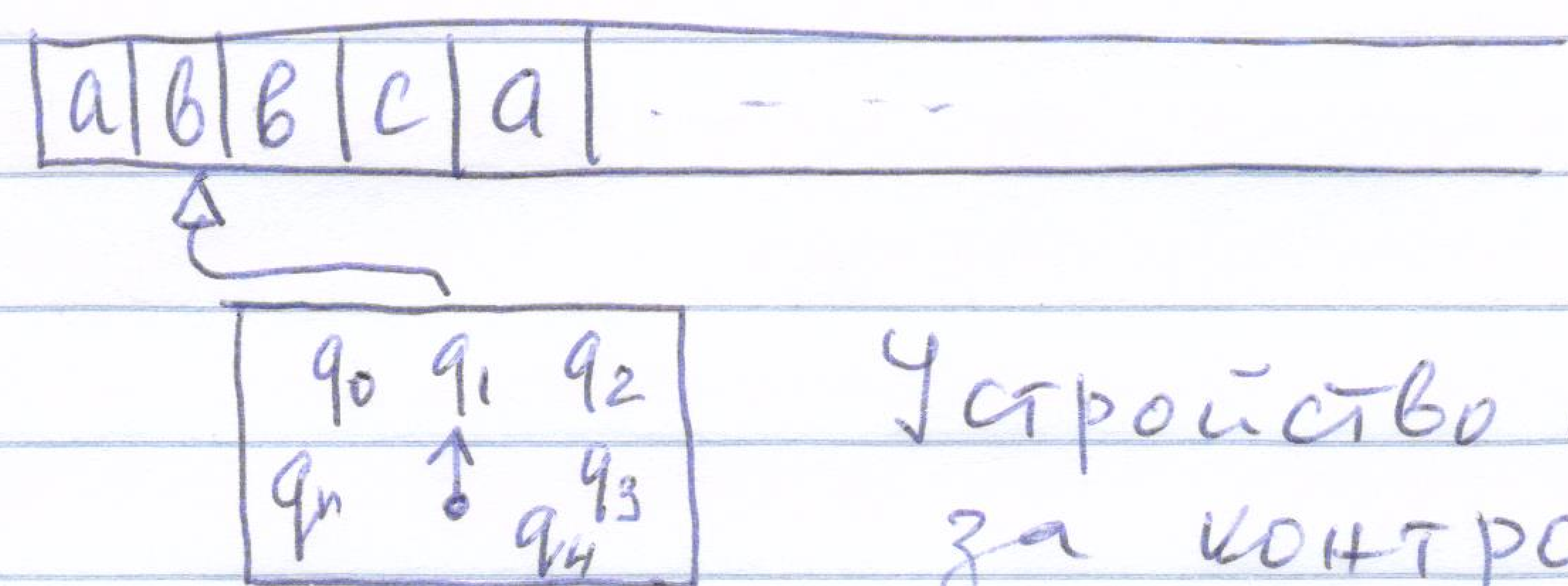
Сега можем и по друг начин да определим регулярните езици, а именно: Класът на регулярните езици в азбуката Σ са точно затварянето на множеството от езиците

$\{\{\sigma\} \mid \sigma \in \Sigma\} \cup \{\phi\}$

относно операциите обединение, конкатенация и звезда на Клини.

Крайни детерминирани автомати

Започваме да се занимаваме с най-простите алгоритмични модели за „пресмятане", известни като крайни детерминирани автомати. Интуитивно, краен детерминиран автомат (КДА) представлява една безкрайна (потенциално) лента с клетки, във всяка клетка на която може да се съдържа само по една буква от дадена азбука. Освен това има една „машина" с четяща глава, която е свързана с устройство с крайно много състояния. Устройството може да променя състоянието си в зависимост от това какво е прочела главата и в какво състояние се е намирало устройството. Картинката можем да я илюстрираме така:



Безкрайна входна лента

Устройство с крайно много състояния за контрол

Формалната дефиниция е следната:

Д. Краен детерминиран автомат (КДА) се нарича петицата $\langle K, \Sigma, \delta, s, F \rangle$ такива, че

K – крайно множество от състояния

Σ – основна азбука, (чийто думи ще разпознаваме)

$s \in K$, s – начално състояние,

$F \subseteq K$, F – множество от заключителни състояния,

$\delta : K \times \Sigma \to K$, δ – функция на преходите.

Интуитивно, най-напред в самото начало имаме думата записана буква по буква в първите няколко клетки. Четящата глава се намира в началото на лентата. Когато главата е в началото, устройството за контрол се намира в началното състояние. В зависимост от това какво е прочела главата и в какво състояние се намира устройството, то преминава в друго състояние, а четящата глава се премества с една клетка надясно. Това продължава, докато четящата глава стигне до края на думата (входа). Тогава в зависимост от това в какво състояние се намира, дали е заключително или не, казваме дали входа се приема или не. За да бъде точно и прецизно реализирано това ще дадем съответните дефиниции.

Д. Нека $M = \langle K, \Sigma, \delta, s, F \rangle$ е КДА. Конфигурация за M се нарича всяка двойка на $K \times \Sigma^*$, т.е. конфигурация е всяка двойка (q, w), където $q \in K$ и $w \in \Sigma^*$. За конфигурациите на M трябва да си мислим като състоянието q в което се намира „устройството за контрол", а за w като непрочетената част от думата. КДА са прости устройства, които нямат памет и главата никога не се връща назад да прочете отново прочетена буква. Ще определим една релация $\vdash_M$ м/у конфигурациите за M.

Д. Нека $(q, w), (q', w')$ са две конфигурации за M. Казваме че конф. (q, w) се преработва за 1 стъпка от M в конф. (q', w') и пишем $(q, w) \vdash_M (q', w')$ т.т.к. когато същ. $a \in \Sigma$ такова, че $w = aw'$ и $\delta(q, a) = q'$.

Всъщност посочената релация $\vdash_M$ е функция, която еднозначно съпоставя на конф. (q, w) конф. (q', w') в зависимост от първата буква на w. Когато стигнем до конф. от вида (q, ε), това означава че сме прочели окончателно входа (входната дума).

С $\vdash_M^*$ означаваме рефлексивното и транзитивно затваряне на релацията $\vdash_M$. Когато $(q, w) \vdash_M^* (q', w')$ ние четем от (q, w) се извежда (q', w') (за много стъпки). Когато $(q, w) \vdash_M^* (q, w)$ ние казваме, че (q, w) се извежда от (q, w) за 0 стъпки, а ако $(q_0, w_0) \vdash_M (q_1, w_1) \vdash_M \dots \vdash_M (q_{n-1}, w_{n-1}) \vdash_M (q_n, w_n)$ то казваме, че (q_n, w_n) се извежда от (q_0, w_0) за n стъпки.

Д. Нека $M = \langle K, \Sigma, \delta, s, F \rangle$ е КДА. Казваме, че думата $w \in \Sigma^*$ се приема (разпознава) от M т.т.к $(s, w) \vdash_M^* (f, \varepsilon)$ и $f \in F$.

С $L(M)$ се означава множ. $\{w \mid w \in \Sigma^*$ и w се разпознава от $M\}$ и $L(M)$ се нарича език разпознаван от M (който се разпознава от M).

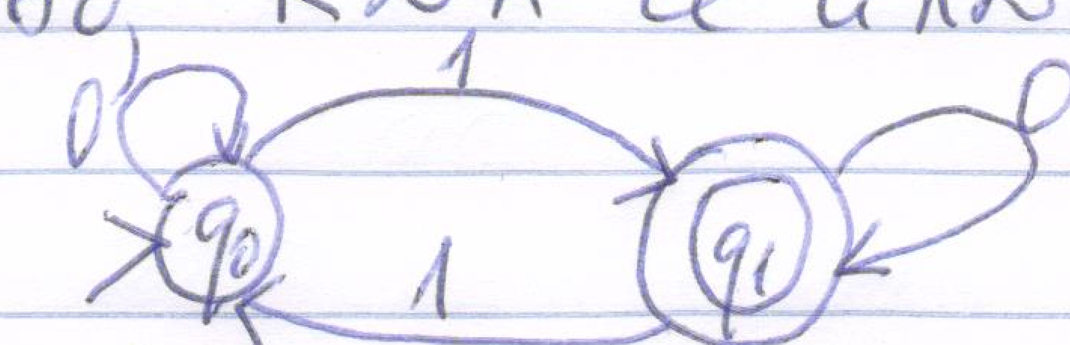
Ще дадем един пример на КДА.

Пример 1. Нека $M = \langle K, \Sigma, \delta, s, F \rangle$, където $s = q_0$, $F = \{q_1\}$, $K = \{q_0, q_1\}$, $\Sigma = \{0, 1\}$, а $\delta : K \times \Sigma \to K$ е зададено с таблицата:

q	σ	$\delta(q, \sigma)$
q_0	0	q_0
q_0	1	q_1
q_1	0	q_1
q_1	1	q_0

Да разгледаме един пример за тази M. Нека $w = 01011$. Тогава $(q_0, 01011) \vdash_M (q_0, 1011) \vdash_M (q_1, 011) \vdash_M (q_1, 11) \vdash_M (q_0, 1) \vdash_M (q_1, \varepsilon)$ т.е. w се разпознава (приема) от M. Лесно може да се види, че $L(M) = \{w \mid w \in \{0,1\}^*$ и w има нечетен брой $1\}$.

Обикновено КДА се илюстрират със следната диаграма (схема):



Ясно е, че M определя напълно тази диаграма и обратно. На упражнения ще видите много и разнообразни примери на автомати и езици разпознавани от тях.