

4.ЗНАНИЯ И ТЯХНОТО ПРЕДСТАВЯНЕ

4.1 Изисквания към формализмите за представяне на знанията.

Представянето на знанията има две страни:

- представяне от гледна точка на съхраняване в паметта (описание) - синтактична страна.
- представяне от гледна точка на преобразуване (обработка) - логическа страна.

Представянето и използването на знанията трябва да се разглеждат в единство.

В основата на всяко представяне е заложен някакъв определен формализъм.

Изискванията към формализмите за представяне на знанията са твърда различни дори в редица случаи противоречиви.

На познавателно равнище изискванията към формализмите за представяне на знанията са следните:

- ☐ естественост по отношение на терминологията и структурата на предметната област;
- ☐ структурираност , колкото се може по-близка до структурата на предметната област;

На логическо равнище (на ниво на обработка) основните изисквания са следните:

- ☐ прозрачност, т.е. ясна семантика (правилно построените изрази трябва да имат ясен смисъл);
- ☐ композиционност, т.е. валидност на принципа на композицията (значението на сложните изрази да се определя напълно от изразите, които ги съставят)
- ☐ изводимост (всяко извлечено по правилата знание от верните знания в системата е също вярно).

На равнище на реализация основните изисквания са:

- ☐ ефективност по време
- ☐ ефективност по памет

4.2.Процедурни и декларативни формализми за представяне на знанията.

При декларативните (описателни) формализми за представяне на знанията основната тежест пада върху представянето.

От двата въпроса “какво представлява?” и “как се прави?” – по-лесно се отговаря на първия.

Характерно за декларативните формализми е следното:

- процедурите за извод са достатъчно общи и работят с различни знания (и в различни предметни области).
- тъй като на преден план е представянето, обновлението се осъществява по-лесно.
- знанията и връзките между тях са разделени.

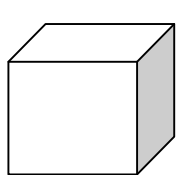
При процедурните формализми (действие) водещо е използването на знанията. Основната тежест е върху въпроса “как?”.

Процедурното знание съдържа знание + начините за неговото използване. Ако тези начини са няколко, то посочени са всичките. Знанията и връзките между тях са заедно в тясно единство.

Характерно за процедурните формализми е следното:

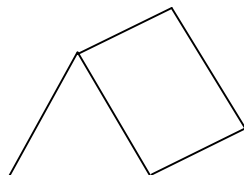
- всички алгоритмични знания са от процедурен характер;
- за да задействат алгоритъма данните трябва да се подадат в подходящ вид;
- процедурните знания имат по-голяма ефективност (и по време, и по памет).

За да сизразим по-релефно различието между тези два формализма, ще приведем един пример от света на кубчетата, в който има само два вида обекти.



Куб

Призма



Да приведем описанието на понятието кула.

Декларативно описание на кула:

“Кулата е подредени едно над друго няколко кубчета, като най-отгоре е поставена призма.”

Процедурно описание на кула.

“Кулата се строй като:

1. поставя се кубче на пода;
2. взема се кубче, над което няма друго кубче;
3. взетото кубче се поставя на върха на кулата;
4. накрая на върха се поставя призма.”

Представянето и използването на знания чрез процедури е твърде широко разпространена практика. Всяка програма представлява процедурно знание. Вариант на това представяне са т.нар. “демони” (Уинстън 1977г.) - процедури, в които са закодирани действията в определени ситуации, и които се активират само при настъпване на конкретна ситуация.

Характерно за такива системи е тяхната **хетероархична структура**.

Знанията се изразяват в семантиката на самите процедури, а връзките между знанията се представят чрез връзки между отделни процедури – чрез обръщението към други процедури и/или предаване на параметри. Всяко процедурно знание е ориентирано към решаването на конкретен проблем.

При използване на процедурно знание се постига висока ефективност, но това е съпроводено със следното:

- щателно планиране на възможните ситуации;
- при всяка ситуация стартира единствена процедура;

- механизъм за действие при ситуации, които стартират повече от една процедура;
- механизъм за действие при поява на непланирана ситуация.

С ограничен набор от процедури може да се обходи достатъчно сложен лабиринт, както е например в мита за Ариадна, изпратила Тезей да убие Минотавъра, който се крие в заплетен лабиринт.

Ариадна дала кълбо на Тезей и хванала края на нишката в ръката си. Тезей, размотава кълбото (и го намотава когато се връща), докато обходил лабиринта и намерил чудовището.

4.3.Представяне и използване на знания чрез процедури.

Нека покажем как едно понятие може да бъде описано чрез алгоритъм. Да означаваме променливите, които управляват процеса с p , q , r и s . В най-простия случай те са двоични.

Действията по снемане на неопределеност да включим в понятието оператор и да ги бележим с A , B , C , D ,

Да въведем (следвайки Шапиро) две форми на процеса на формиране на понятия:

- операторна и
- логическа.

Операторната форма е свързана със самия процес за формиране на понятието

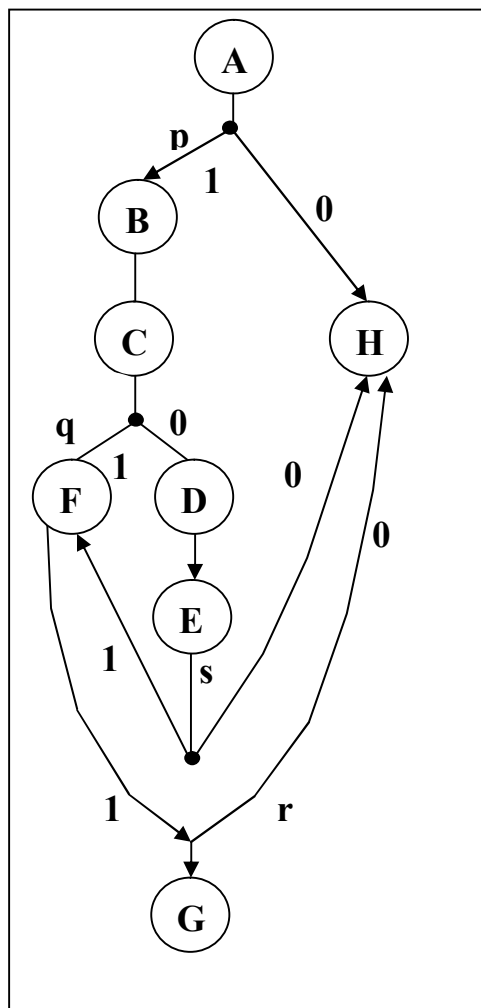
Логическата форма е свързана със завършената структура на понятието.

Пример: *Алгоритъм за формиране на понятието височина на триъгълник.*

Да отворим учебника и да прочетем дефиницията за височина на триъгълник.

“Отсечката а е височина на триъгълник ΔABC , ако:

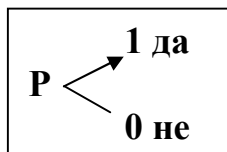
- а) един от нейните краища е връх на ΔABC ($p=1$);
- б) вътрешният ѝ край лежи или на противоположщата страна на Δ ($q=1$), или върху нейното продължение. ($s=1$)
- в) тя е $\perp$ към направлението на противоположщата на върха страна на Δ ($r=1$).”



$Ap \uparrow^1 \quad BCq \uparrow^2 \downarrow^3 \quad Fr \uparrow^1$
 $G. \downarrow^2 \quad DE \quad \bar{s} \quad \uparrow^3 \downarrow^1 \quad H$
операторна форма

Графичен запис

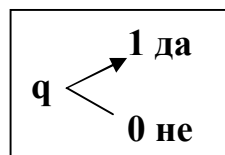
A - проверка дали
единият от краищата на
отсечката е връх на
триъгълника; да го наречем
именован връх.



H - извод – отсечката
не е височина на
триъгълника.

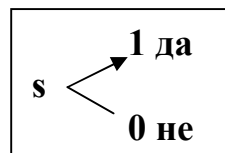
B - намиране на
противолежщата на
върха(именования) страна
на триъгълника.

C - проверка дали
другия край на отсечката
лежи върху
противолежщата на
триъгълника страна; да я
наречем именувана страна.

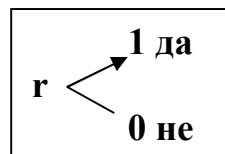


D – намира се продължението на именованата страна.

E – проверка дали другият край на отсечката принадлежи на
продължението.



F – проверка за перпендикулярност между отсечката и
(именованата) противоположаща страна на триъгълника.



G – заключение – отсечката е височина на триъгълника.

Нека запишем таблицата на истинност за логическата
функция, която съответствува на обходените при всяко възможно
изпълнение оператори (определят се от възможните комбинации от
стойности на логическите променливи от алгоритъма).

N	p	q	r	s	Изпълнявана последователност от оператори	f
0	0	0	0	0	AH	0
1	0	0	0	1	AH	0
2	...	...	...	...	...	...
7	0	1	1	1	AH	0
8	1	0	0	0	ABCDEH	0
9	1	0	0	1	ABCDEFH	0
10	1	0	1	0	ABCDEH	0
11	1	0	1	1	ABCDEFGF	1
12	1	1	0	0	ABCFH	0
13	1	1	0	1	ABGFH	0
14	1	1	1	0	ABCFG	1
15	1	1	1	1	ABGFG	1

Ние описахме понятието “височина на триъгълник” чрез средства, които са външни към самото понятие.

Описахме ги на операторен език. Но къде отиде смисълът?! Как по описание, което въобще не засягаше въпроса за смисъла, може да се съди за същността (смисъла, семантиката) на понятието височина на триъгълника?

Нека спрем вниманието си върху особеностите на двете форми на понятието.

Операторната форма има:

а) съдържание и структура.

Нейното съдържание са включените в нея оператори (A,B, C, D, E, F, G, H), а нейната структура са логическите й условия и съответната й функция (p, q, r, s и f).

б) голямо количество еквивалентни форми:

Например:

$BC \overset{\uparrow 1 \downarrow 2}{q} \overset{\uparrow 3}{F} \overset{\downarrow 1}{r} \overset{\uparrow 2 \downarrow 3}{DE} \overset{\uparrow 3}{H} \overset{\downarrow 1}{A} \overset{\uparrow 3}{p}$

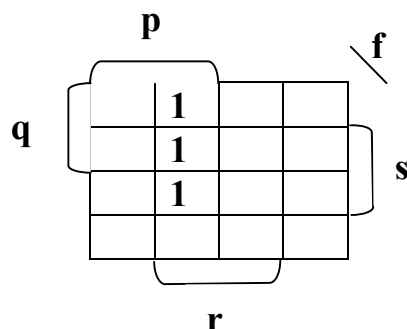
$\overset{\downarrow 1}{F} \overset{\uparrow 2 \downarrow 3}{r} \overset{\uparrow 3}{H} \overset{\downarrow 1}{A} \overset{\uparrow 3}{p} \overset{\downarrow 1}{BC} \overset{\uparrow 1}{q} \overset{\uparrow 2 \downarrow 3}{DE} \overset{\downarrow 1}{s} \overset{\uparrow 3 \downarrow 2}{G}$

За да установите, че тези форми са еквивалентни, необходимо е да ги сравните, а това съвсем не е лесна работа. Нека добавим и възможността да се обединят последователности от оператори в операторна група.

$BC \rightarrow R$ и $DE \rightarrow Q$

R и Q са нови оператори.

В резултат от обединението се



$$f = pqr \vee psr = pr (q \vee s)$$

Логическа форма за понятието височина на триъгълник

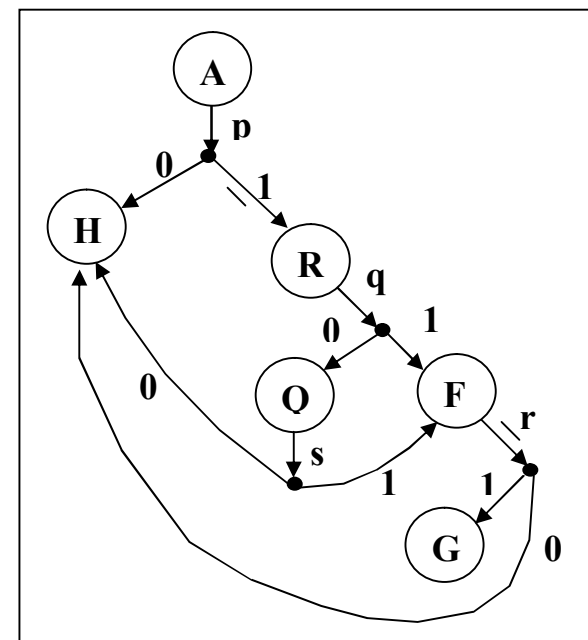
И така, при изобразяване на понятието чрез алгоритмична структура могат да се открият две форми:

-операторна форма на понятието,

-логическа форма на понятието.

И двете форми по същество са правила да се реши задачата дали дадена отсечка е височина на триъгълника.

Обърнете внимание, че операторът освен действие съдържа в себе си и своето следствие (той при всяко свое изпълнение посочва и кой оператор да го следва.)



получи ново съдържание, а формата остана същата.

$$A p^{\uparrow} R q^{\uparrow 2 \downarrow 3} F r^{\uparrow 1} G. \downarrow 2 Q_s^{-\uparrow 3 \downarrow 1} H.$$

Интересно е, че различните форми имат едно и също изчертаване в графична форма. Следователно удобното за компютъра стрингово изобразяване внася допълнително “объркване” и квазиразнообразие.

Може би силата на човешкото мислене е във възможността да се работи с двумерни (а защо не многомерни) описания на обектите.

Компютрите ще направят качествен скок в своето развитие, когато излязат извън тесните рамки на стринга!

На първо време поне по-активно да използват логическата форма. Тя е интересна и твърде важна като есенция на понятието, тя е скелет на отговора да!

Изграждането на система с процедурно знание може да следва следната схема:

- основните понятия в предметната област се описват чрез разпознаващ ги алгоритъм;
- от тези понятия и следвайки правилата на композицията в съответствие с ролята им в предметната област се строят нови (производни) понятия;
- всяко понятие ще има съдържание (набор от оператори) и форма (логическа функция).

В такава система може освен елементарната проверка за принадлежност на обект към дадено понятие, да се изпълняват и по-сложни неща. Например може да се търсят аналогии

- а) по верига от оператори,
- б) по логическа форма,
- в) по група условия,
- г) по операторно съждение.

Обучението на подобна система би могло да се осъществи в разнообразни форми, като например:

- 1) чрез директно задаване на понятия,

- 2) чрез задаване на част от формата и съдържанието
- 3) обучение чрез аналогия и т.н.

4.4. Предимства и недостатъци на процедурните формализми.

Основен проблем при процедурните формализми е внасянето на изменения в базата знания..

Причина за това е фактът, че връзките между знанията са твърдо заложени в процедурите (поне чрез изработката на параметри за ситуация)

Добавянето на нова процедура означава

* промяна във всички съществуващи вече, които ще се свързват с нея.

** или анализ на всички ситуации по отношение на новата процедура (кога се задейства, дали не се задейства заедно с други и т.н.)

Основно предимство на процедурните формализми е по-висока ефективност и бързодействие. Управлението се предава директно, без да се налага търсене в базата знания.

Процедурните формализми са удобни при моделиране на процеси. Текущото състояние на процеса задейства онази процедура, която “знае” какво следва (не е необходимо да се търси в цялата базата знания).

Основните недостатъци на използването на процедурните формализми, освен споменатите трудности при обновяване на знанията, са следните:

- трудно управление на процеса отвън;
- в конкретен момент управлението се поема от дадена процедура – липсва общ поглед върху целия процес;
- трудно се реализират системи за обяснение на получения резултат.

Най-широко се използват там където преобладават алгоритмичните знания (управление на процеси, управление на работи и др.)

Интересни са техните възможности за моделиране на олективно поведение.

Пример: *Моделиране на играта “живот”.*

Играта се развива в двумерно поле от еднакви клетки.

Всяка клетка е свързана с всички съседни.

Алгоритъм: В следващото поколение остават живи (точката остава) или оживяват (запис на точка) онези □, които имат 3 и по-малко живи съседни.

.		.
	.	
		.