

ÚLOHY, ALGORITMY, MODELÝ

II. Sešit série MTM

Stanislav Míka, Ludvík Vlček

Plzeň 2015

Anotace

Ve výkladu se pokoušíme rozbourat (i když spíše nesměle) tradiční strukturu poznatků z matematických učebnic, abychom více zdůraznili vazbu matematiky na problémy praxe a poněkud potlačili školskou hierarchii výkladu. U každé kapitoly je také uveden obsah kapitoly, aby se čtenář mohl rozhodnout, zda si ji přečte, nebo ji „přeskočí“.

Ve výkladu se také „dopouštíme“ prohrěšku stručnosti tím, že často opakujeme část z předchozího výkladu. To je diktováno snahou ulehčit čtenáři studium textu.

V první kapitole informujeme čtenáře o správné formulaci matematických úloh. Zde zdůrazňujeme důležitost vstupních a výstupních dat úlohy. Zmiňujeme se také o řešitelnosti úloh. Také uvádíme jakýsi všeobecně vzdělávací seznam těch nejzajímavějších úloh z historie.

Druhá kapitola je věnována podrobnějšímu výkladu problematiky algoritmů a to nejen z úzce matematického pohledu. Výklad obsahuje přehled hlubších výsledků z teorie složitosti.

Ve třetí kapitole se věnujeme základnímu typu matematických úloh, a to rovnicím. Tato kapitola je těžištěm matematické části. Jasně rozlišujeme pojmy rovnice a rovnosti.

Čtvrtá kapitola je jakýmsi pokusem uvést čtenáře do problematiky matematického modelování. Vybíráme takové „modelované jevy“, u nichž vystačíme s jednoduchými matematickými prostředky a nástroji.

Závěrečná kapitola zaznamenává nejen názory a zkušenosti autorů, ale i některé historické souvislosti a rozšiřující poznatky.

Dovolíme si ještě citovat několik vět z přednášky prof. Petra Piňhy, přednesené 23. 9. 2010 na konferenci Jak připravit učitele matematiky, pořádané katedrou didaktiky matematiky MFF UK:

Žijeme v době nedostatku přesného a úplného splnění úkolů (úloh). V životě pak narážíme na šlendrián a nekončící výmluvy, že v podstatě je vše hotovo a v pořádku, pouze se vloudila chybička, nebo posunul termín. Matematická úloha je případem, kdy takto postupovat nejde. Příklad není vypočítán (úloha není vyřešena) tím, že žák ví, jak na to. Know how musí být aplikováno a realizováno. Matematika je snad jediný předmět, kde nikoho nenapadlo zadat test s otázkami typu:

Je $2 \cdot 17 = 34$ nebo 36 nebo 0 nebo 289 – označte i více správných odpovědí.

Velké heslo musíme děti naučit myslet svedlo mnoho učitelů v mnoha předmětech k tomu, že učí děti, jak se hádat a prosazovat svůj nesmysl.

Matematika vyžaduje znalosti a kázeň, bez nichž žádné pořádné myšlení neexistuje. Vyžaduje také, díky tomu, že jde o teorii, znalost teorémů a axiomů a všeho, na čem stojí vyšší patra daného oboru.

V matematice na rozdíl od některých jiných předmětů není přípustné, ani možné, aby žák řešil diferenciální rovnici bez znalosti sčítání či násobilky. Máme dnes v jiných oborech množství odborníků, kteří *mutatis mutandis* neumí jejich násobilku.

Obsah II. sešitu

	Strana
1 Úlohy	7
1.1 Formulace matematických úloh	8
1.1.1 Úlohy a příklady	8
1.1.2 Řešitelnost úloh	8
1.2 Slavné i zajímavé úlohy	9
1.2.1 Problémy dodatečného Eukleidova postulátu	9
1.2.2 Diofantovy úlohy a diofantické rovnice	10
1.2.3 Izoperimetrické úlohy - Úlohy princezny Didó	12
1.2.4 Úloha sedmi mostů města Královce	13
1.2.5 Matematické problémy tisíciletí	14
1.2.6 Hilbertovy problémy	16
2 Algoritmy	19
2.1 Co je to algoritmus?	19
2.1.1 Různá pojetí algoritmu	20
2.1.2 Reprezentace a zápis algoritmu	21
2.2 Obecné vlastnosti algoritmů	22
2.3 Způsoby klasifikace algoritmů	23
2.3.1 Dělení algoritmů	24
2.3.2 Zpracování jednotlivých kroků algoritmu	25
2.3.3 Nároky na zdroje (potřeba zdrojů)	25
2.3.4 Třídy složitosti	26
2.4 Časová složitost algoritmů	27
2.4.1 Časová složitost nalezení maxima	27
2.4.2 Časová složitost algoritmu <i>bubblesort</i>	28
2.4.3 Ostatní časové složitosti	29
2.5 Asymptotická složitost algoritmů	30
2.5.1 Asymptotická složitost algoritmu <i>bubblesort</i>	30
2.5.2 Asymptotická složitost algoritmu <i>quicksort</i>	31
2.5.3 Problémy asymptotické složitosti	33
3 Zákon přírody - symboly poznání	35
3.1 Rovnice a rovnosti	36
3.1.1 Výlet do školních lavic	36
3.1.2 Lineární (algebraická) rovnice pro jednu neznámou	37
3.1.3 Polynomiální rovnice pro jednu neznámou	38
3.1.4 Obecná rovnice pro jednu neznámou	39
3.2 Jedna rovnice pro více neznámých	41

3.2.1	Jedna rovnice pro dvě neznámé	41
3.2.2	Funkcionální rovnice - globální a lokální řešení	43
3.3	Rovnice z geometrie	45
3.3.1	Rovnice křivek a ploch	45
3.3.2	Soustava dvou rovnic pro dvě neznámé. Grafický a	47
3.4	Rovnice lineární algebry	48
3.4.1	Soustava dvou rovnic pro tři neznámé	48
3.4.2	Základní úloha lineární algebry	50
4	Matematické modely některých zákonů	53
4.1	Zákony ekonomické rovnováhy	54
4.1.1	Zákon nabídky a poptávky	54
4.1.2	Model uzavřené ekonomiky	58
4.2	Zákony mechanické rovnováhy	60
4.2.1	Archimédův zákon	60
4.2.2	Stavové zákony	61
5	Hovory M	65
5.1	Školské formulace úloh	65
5.2	Eliminační principy	66
5.2.1	Podivuhodná role expertů na základní škole	66
5.2.2	Determinanty a další starožitnosti	67
5.3	Opět Al-Chwárizmí	67
5.3.1	Historie evropské (nejen matematické) vzdělanosti	67
5.3.2	Díla Al-Chwárizmího	69
5.4	Celý svět modeluje	70
5.4.1	Doby stárnou rychleji než lidé	73
5.4.2	Matematické modelování a vzdělávání modelářů	73
5.4.3	„Antimodelářská“ výuka matematiky	76
5.5	Podrobnosti o slavných úlohách	76
5.5.1	Pověst o založení Kartága	76
5.5.2	Brachystochrona nebo brachistochrona	78

Kapitola 1

Úlohy

Obsah kapitoly

1.1	Formulace matematických úloh	8
1.1.1	Úlohy a příklady	8
1.1.2	Řešitelnost úloh	8
1.2	Slavné i zajímavé úlohy	9
1.2.1	Problémy dodatečného Eukleidova postulátu	9
1.2.2	Diofantovy úlohy a diofantické rovnice	10
1.2.3	Izoperimetrické úlohy - Úlohy princezny Didó	12
1.2.4	Úloha sedmi mostů města Královce	13
1.2.5	Matematické problémy tisíciletí	14
1.2.6	Hilbertovy problémy	16

Motto: Příliš mnoho příliš nudných nic neříkajících rovnic

*(stanovisko studenta FAV ZČU na anketní hodnocení
přednášky z parciálních diferenciálních rovnic)*

Vycházíme z předpokladu, že obsah pojmu „úloha“ (úkol, problém, ...) je čtenáři alespoň částečně znám a že cítí souvislosti s termíny „formulace úlohy“, „zadání úlohy“, „data úlohy“, „řešení úlohy“, „řešitelnost úlohy“, „výsledek“, ... atd.

Řešíme-li nějaký problém (úlohu), hledáme odpovědi na dané nebo formulované otázky. Pokud lze úlohu formulovat matematickými prostředky, hovoříme o matematické úloze.

Abychom například mohli stanovit vývoj (prognózu) nějakého systému, musíme si nejdříve vytvořit matematický model systému. Ten nám pak umožní formulovat matematickou úlohu.

1.1 Formulace matematických úloh

1.1.1 Úlohy a příklady

Matematická úloha a její formulace musí obsahovat

- a) *vstupní informace* (data úlohy, zadání, „co je dáno/známo“, ...),
- b) *výstupní informace* (výstupní data, požadovaný výsledek, „co se hledá“, cíl, co se má najít).

Hledání vztahů mezi vstupními a výstupními daty se říká *analýza (rozbor) úlohy*. V řeči politiků se pro analýzu používá slovní spojení „řešíme problém“.

Důležité upozornění: V českém jazyce má termín *řešení úlohy* dva významy:

- a) proces (angl. *solving*) - směřující k získání výsledku,
- b) výsledek procesu z bodu a) - (agl. *solution*).

Termínem „příklad“ se v učebnicích označuje úloha, která má ilustrativní nebo procvičovací charakter. Cílem bývá obvykle ozřejmit nějaké pojmy nebo procvičit nějaké (výpočetní) postupy.

1.1.2 Řešitelnost úloh

Řešitelnost úloh je (zjednodušeně řečeno) soubor odpovědí na otázky typu:

- za jakých podmínek existuje řešení (jedno, více, nekonečně mnoho) dané úlohy.

Abychom mohli formulovat otázky i odpovědi, musíme budovat systém poznatků, který obvykle nazýváme *teorie řešitelnosti* resp. *teorie úlohy*.

Např. teorii řešitelnosti úlohy stanovit k dané funkci její integrál je teorie Riemannova integrálu nebo teorie Lebesgueova integrálu.

Korektní a nekorektní úlohy.

Řekli jsme, že matematické úlohy (nejen numerické) lze obecně chápat jako funkční vztah mezi vstupními daty a výstupními daty (výsledky) typu:

$$\mathbf{w} = U(\mathbf{v}), \mathbf{v} \in \mathcal{B}_1, \mathbf{w} \in \mathcal{B}_2,$$

kde: $\mathcal{B}_1$ je množina obsahující vstupní data,

$\mathcal{B}_2$ je množina obsahující výstupní data.

Řekneme, že daná úloha je **korektní** (stabilní) na dvojici množin $(\mathcal{B}_1, \mathcal{B}_2)$, když má tyto dvě vlastnosti:

1. ke každému $\mathbf{v} \in \mathcal{B}_1$ existuje jediné $\mathbf{w} \in \mathcal{B}_2$,
2. jsou-li $\mathbf{w} = U(\mathbf{v})$, $\mathbf{w}_a = U(\mathbf{v}_a)$ výsledky pro data $\mathbf{v}, \mathbf{v}_a \in \mathcal{B}_1$, pak existuje konstanta K , že platí:

$$\|\mathbf{w} - \mathbf{w}_a\| \leq K \cdot \|\mathbf{v} - \mathbf{v}_a\|$$

Vlastnost 2 se také nazývá *spojitá závislost řešení na vstupních datech*. Znamená to, že když $(\mathbf{v} - \mathbf{v}_a)$ je „malé“, potom $(\mathbf{w} - \mathbf{w}_a)$ je také (srovnatelně) malé.

Omlouváme se za příliš „nážornou“ definici.

Příkladem nekorektní úlohy je úloha najít reálné kořeny polynomu:

$$p(z) = z^4 - z^2(2a - 1) + a(a - 1).$$

- pro $a \geq 1$ má polynom 4 reálné kořeny,
- pro $0 \leq a < 1$ má polynom 2 reálné kořeny
- pro $a < 0$ nemá žádný reálný kořen

tj. při malé změně vstupního parametru a v množině reálných čísel dochází ke skokové změně počtu reálných kořenů.

Úlohy, které nesplňují podmínky 1 a 2 se nazývají *nekorektní* (nestabilní).

Velkou řadu nekorektních úloh tvoří *nejednoznačně řešitelné* úlohy.

Např. pro úlohu:

$$\mathbf{Ax} = \mathbf{b} \quad (\text{soustava lineárních algebraických rovnic})$$

můžeme výše deklarovaný vztah vstupních a výstupních dat zapsat ve tvaru:

$$\mathbf{x} = U(\mathbf{A}, \mathbf{b}),$$

a jenom pro korektní (jednoznačně řešitelnou) úlohu může být tento vztah zapsán jako:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}.$$

1.2 Slavné i zajímavé úlohy

1.2.1 Problémy dodatečného Eukleidova postulátu

Eukleidovy postuláty (Eukleidovy axiomy), na základě kterých axiomaticky vystavěl geometrii ve svých *Základech*, jsou:

1. První postulát: budiž úkolem od kteréhokoliv bodu ke kterémukoliv vésti přímku. (Dvěma body lze vést přímku.)
2. Druhý postulát: a přímku omezenou nepřetržitě rovně prodloužiti. (Přímku lze neomezeně prodloužovat. Pod pojmem přímka se původně rozuměl geometrický útvar, který v současné době nazýváme úsečka.)
3. Třetí postulát: a z jakéhokoli středu a jakýmkoli poloměrem narýsovat kruh. (Ze zadaného středu lze opsat kružnici se zadaným poloměrem.)
4. Čtvrtý postulát: a že všechny pravé úhly sobě rovny jsou. (Všechny pravé úhly jsou stejné.)

5. Pátý postulát: a když přímka protínající dvě přímky tvoří na téže straně vnitřní (přilehlé) úhly menší dvou pravých, ty dvě přímky prodlouženy jsouce do nekonečna že se sbíhají na té straně, kde jsou úhly menší dvou pravých.

Dva tisíce let se učenci snažili podat důkaz, že pátý Eukleidův postulát je důsledkem prvních čtyř. Tento postulát je totiž výrazně složitější než postuláty zbylé, a to nejen svým zněním ale také významem - nepopisuje totiž žádnou fundamentální vlastnost základních geometrických objektů ale je spíše jistým netriviálním tvrzením o nich. Výsledkem těchto neúspěšných pokusů o důkaz je celý seznam vět, které jsou ekvivalentní s pátým postulátem (tj. mohou jej nahradit). Mezi ně patří například věta „součet vnitřních úhlů v trojúhelníku je roven dvěma pravým“ nebo *Pythagorova věta*.

Všechny pokusy o důkaz tohoto postulátu byly ukončeny začátkem 19. století (Lobačevskij, Bolyai, Gauss) s negativním závěrem, že tento postulát je v rámci tzv. eukleidovské geometrie nedokazatelný.

Pokud tedy platnost tohoto postulátu nepředpokládáme, resp. nahradíme jej jiným axiomem, dostaneme jinou geometrii (!).

Stála ta námaha dvou tisíce let za to?

Ano!

Einsteinova obecná teorie relativity se bez „jiné“ geometrie neobejde. Momentální výsledky tohoto snažení („řešení problémů“) zatím končí u *Gödel*a a současném pohledu na axiomatické struktury matematiky.

1.2.2 Diofantovy úlohy a diofantické rovnice

Tyto úlohy lze považovat za začátek vývoje teorie čísel, algebraických struktur s důsledky do současnosti. Viz závěr odstavce.

 **Úloha o třech krychlích:** (Diofantos z Alexandrie, asi 200-284 n.l.)

Pro libovolnou krychli o straně z stanovit dvě krychle o stranách x, y takové, aby platilo:

$$x^3 + y^3 = z^3.$$

Obecně: Najít přirozená čísla x, y, z a n , pro která platí:

$$x^n + y^n = z^n, \text{ kde } n > 2.$$

Víme, že pro: $n = 1$ má rovnice $x + y = z$ o třech neznámých nekonečně mnoho řešení;

pro: $n = 2$ má rovnice $x^2 + y^2 = z^2$ nekonečně mnoho řešení, např. každý přirozený násobek trojice $(3, 4, 5)$ - pythagorejské trojice.)

V 17. století si francouzský matematik *Pierre de Fermat* (1601-1665) poznamenal na okraj jedné knihy toto:

Je nemožné rozdělit krychli do dvou krychlí, či čtvrtou mocninu do dvou čtvrtých mocnin, nebo obecně jakoukoli mocninu vyšší než druhou do dvou stejných mocnin.

Objevil jsem opravdu tak podivuhodný důkaz, že tento okraj je příliš malý, aby se do něj vešel.

Uvedený důkaz ovšem nebyl v jeho pozůstalosti objeven – víme, že Fermat našel důkaz pro n rovno čtyřem, ale nejspíše nikoli pro jiné exponenty.

Tuto větu (*Velká Fermatova věta*) se pokoušeli matematici dokázat déle než 300 let. Mezi neúspěšnými řešiteli problému nalezneme slavná jména jako byl *Euler*, *Lamé*, *Cauchy*.

Dokázat tuto větu se povedlo až v roce 1994 britskému matematikovi Andrew Wilesovi, a to prostředky matematiky 20. století. Samotný důkaz má zhruba 100 stran a nepředpokládá se, že by mohl existovat nějaký jednoduchý způsob, jak větu dokázat, jak tvrdil Pierre de Fermat.

Velká Fermatova věta je jedna z nejslavnějších vět v historii matematiky. Zní takto:

Neexistují přirozená čísla x, y, z a n , pro která $x^n + y^n = z^n$, kde $n > 2$ a $x, y, z \neq 0$.

Úloha z Diofantova náhrobku:

Bůh mu dopřál, aby:

byl hochem šestinu svého života a přidal k této době další dvanáctinu, ozdobil jeho líce vousem. Po další sedmině prozářil jeho život světlem manželství, po dalších pěti letech pak daroval mu syna. Však běda! Sotva ubohé dítě dosáhlo poloviny délky otcova života, neúprosné sudičky vzaly si jej zpět. Když Bůh utěšil jeho hoře učením o číslech, po dalších čtyřech letech ukončil dobu jeho žití.

Můžeme prozradit, že Diofantos zemřel ve věku 84 let. V jazyku rovnic dostaneme:

$$\frac{1}{6}x + \frac{1}{12}x + \frac{1}{7}x + 5 = \frac{1}{2}x + 4$$

Úloha rybářů a Diracovo řešení:

Tři rybáři společně ulovili určité množství ryb s tím, že jejich podíly na úlovku jsou třetinové. Pak ulehli ke spánku.

 ~~Pak~~ první se vzbudil jeden z nich a chtěl si odnést svůj podíl. Počet ryb ale nebyl dělitelný třemi, proto jednu rybu pustil zpět do vody. Vzal si třetinu zbývajících počtu a odešel. Když se vzbudil druhý rybář, situace se opakovala. Jednu rybu pustil, vzal si třetinu a odešel. Totéž udělal třetí rybář.

Otázka je, kolik bylo ulovených ryb a kolik jich zbylo po tomto dělení se zbytkem.

Úloha je poněkud absurdní, ale rybáři neuměli zlomky a nedovedli si představit, že by rybu rozsekali na části.

Úlohu můžeme přeformulovat jako bilanci počtu ryb. Označíme:

- počet ulovených ryb: x
- počet zbylých ryb: y

Po úpravě dostaneme rovnici:

$$8x = 27y + 38$$

(linární algebraická rovnice pro dvě neznámé).

Jedna dvojice výsledků je:

$$x = 25, y = 6.$$

Zjistěte aspoň těch „zbylých“ nekonečně mnoho dvojic přirozených čísel.

Když tuto úlohu ve škole řešil slavný fyzik *Paul Dirac* (1902-1984), prohlásil, že rybáři chytili -2 ryby. Z hlediska rybářů je odpověď absurdní, ale toto číslo je řešením příslušné rovnice. Navíc pěkným, protože je jediné, při němž také $y = x = -2$.

Později Dirac jako první předpověděl existenci antihmoty, když fyzikálně interpretoval podobně „absurdní“ řešení *Schrödingerovy rovnice*.

1.2.3 Izoperimetrické úlohy - Úlohy princezny Didó

Mezi všemi rovinnými oblastmi ohraničené uzavřenými křivkami předepsané délky máme najít takovou křivku, jež ohraničuje oblast maximálního obsahu. Analogicky, mezi všemi prostorovými oblastmi ohraničené uzavřenými plochami předepsaného povrchu máme najít takovou plochu, která ohraničuje oblast s maximálním objemem.

Současnou verzí těchto úloh jsou tzv. *úlohy tvarové optimalizace*. V průmyslové aerodynamice se řeší úlohy nalezení takového aerodynamického profilu (křídla letadla, lopatky turbin), aby ztráty výkonu byly minimální.

Není třeba zdůrazňovat, že jsou to úlohy velmi náročné jak na matematickou teorii, tak na numerické metody a algoritmy, včetně vysokého výkonu počítačů.

a) První Didonina úloha

Mezi všemi oblouky AB délky l_D ležícími v polorovině ohraničené přímkou p , pevným koncovým bodem $A \in p$ a pohyblivým bodem $B \in p$, najít takový oblouk, který spolu s úsečkou AB ohraničuje obrazec největšího obsahu (viz obrázek 1.2.1).

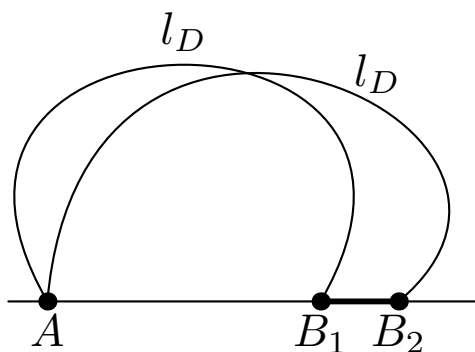
b) Druhá Didonina úloha

Mezi všemi oblouky délky l_D ležícími v polorovině ohraničené přímkou p a **danými** (pevnými) body $A, B \in p$, najít takový oblouk, který spolu s úsečkou AB ohraničuje obrazec největšího obsahu (viz obrázek 1.2.2). Můžeme prozradit, že hledaným obloukem je část kružnice.

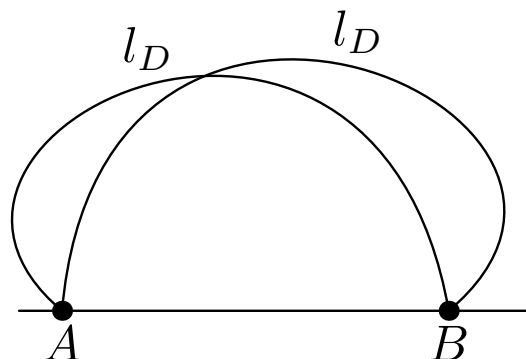
c) Třetí Didonina úloha

Mezi všemi oblouky ležícími v polorovině ohraničené přímkou p a danými (pevnými) body $A, B \in p$ najít takový oblouk (křivku), který má pro daný obsah spolu s úsečkou AB nejkratší délku.

Zde budeme obecně formulovat úlohu tohoto typu. Konkrétnější verzi uvedeme v sešitu věnovaném optimalizačním technologiím.



Obrázek 1.2.1: Možné tvary oblasti ohraničené danou délkou oblouku l_D



Obrázek 1.2.2: Možné tvary oblasti ohraničené danou délkou oblouku l_D

Oblast v rovině označíme Ω a její hranici $\partial\Omega$. Obsah (míru) oblasti Ω označíme $\mu(\Omega)$. Dovedeme si představit, že mezi všemi možnými obdélníky Ω s daným obvodem (délkou hranice) chceme najít takový, který má největší obsah.

Délku hranice označíme $l(\partial\Omega)$.

Formulace úlohy:

Z množiny $\{\Omega\}$ všech oblastí Ω s délkou hranice splňující rovnost (vazbu) $l(\partial\Omega) = 1$ najít takovou oblast $\bar{\Omega}$, aby $\max \mu(\Omega) = \mu(\bar{\Omega})$.

Zde se setkáváme s tzv. *funkcí oblasti*, tj. funkcí, jejímž argumentem („nezávisle proměnnou“) je oblast.

Bližší pozornost této úloze bude věnována v sešitu věnovaném optimalizačním technologiím.

1.2.4 Úloha sedmi mostů města Královce

Sedmi mosty města Královce¹ to začalo a v současnosti je to moderní, bouřlivě se rozvíjející se teoretický i aplikační obor. Jde o *diskrétní matematiku* a *teoretickou informatiku*.

Diskrétní matematika se zabývá těmi oblastmi matematiky, kde hrají hlavní úlohu celá čísla a jednotlivé konečné objekty. Diskrétní objekty jsou prezentovány převážně pomocí konečných grafů a množin.

Slovo „diskrétní“ je v názvu míněno jako opak „spojitého“. Obsahuje: Kombinatoriku, Teorii čísel, Teorii grafů, Teorii her, Teorii složitosti, Teorii vyčíslitelnosti. Teoretická informatika je oblast matematické informatiky, která zkoumá matematické zákonitosti, které mají využití v počítačích a zpracování informací. Jsou to (kromě výše zmíněných): Automaty a gramatiky, Umělá inteligence, Teorie informace, Gramatiky a jazyky, Kryptografie, Formální sémantika, Neuronové sítě, Teorie počítačů, Teorie typů, Diskrétní optimalizace.

Město Královec (Königsberg, Kaliningrad) se rozprostírá na březích a ostrovech řeky Pregely. Mezi jednotlivými břehy a ostrovy vede sedm mostů. Obyvatele města zajímalo (začátek 18.století), zda-li mohou při svých promenádách projít všechny mosty takovým způsobem, že se dostanou opět do původního místa, aniž by jednu spojnici přešli dvakrát.

Obrátili se na *Leonharda Eulera*, slavného matematika, a ten jim v roce 1736 sdělil, že to nelze, tj. že zadaná úloha je neřešitelná. Pokud chtějí vyjít z jednoho místa a vrátit se na totéž místo, a procházet každý most pouze jednou, musí mostů být buď více nebo méně, aby možných cest byl sudý počet. Položil tím základ teorie grafů a také (viz odst. 5.5.2).

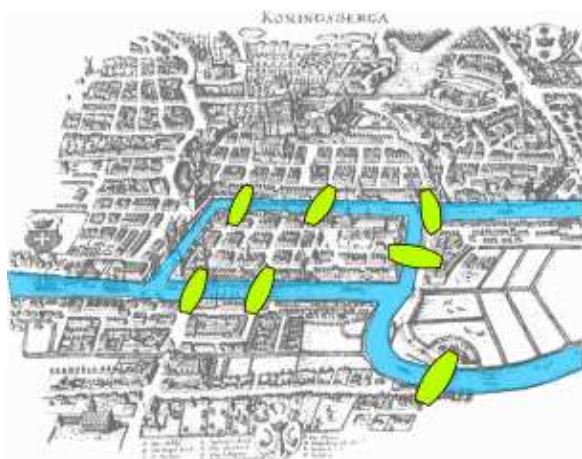
V matematickém jazyce:

Tvrzení 1: Neorientovaný graf lze pokrýt jedním tahem právě tehdy, pokud je souvislý a je Eulerův (má všechny uzly sudého stupně).

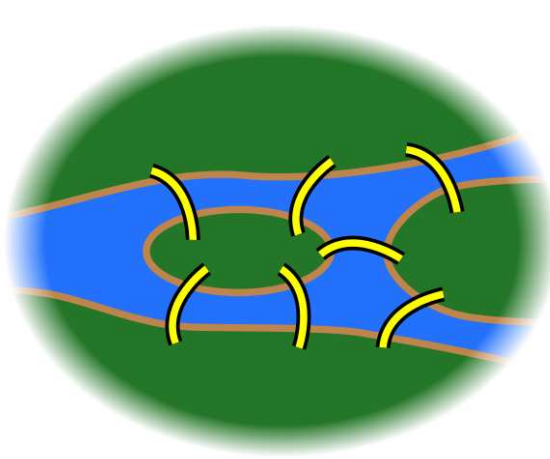
Tvrzení 2: Necht G je souvislý neorientovaný graf, který obsahuje právě $2n$ uzlů lichého stupně (n větší nebo rovno 1), pak se každé jeho minimální pokrytí skládá z n otevřených tahů, z nichž každý spojuje dvojici uzlů lichého stupně.

¹citováno z veřejně dostupných zdrojů

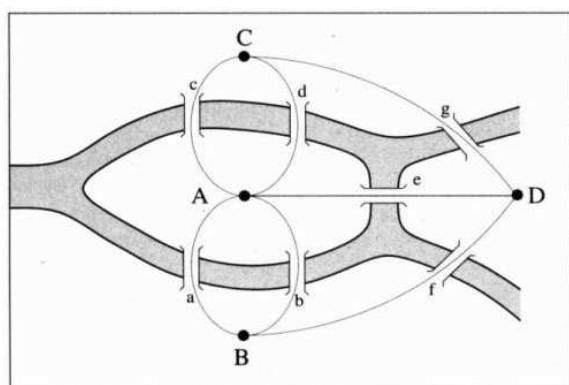
Přiložené obrázky nám napoví více:



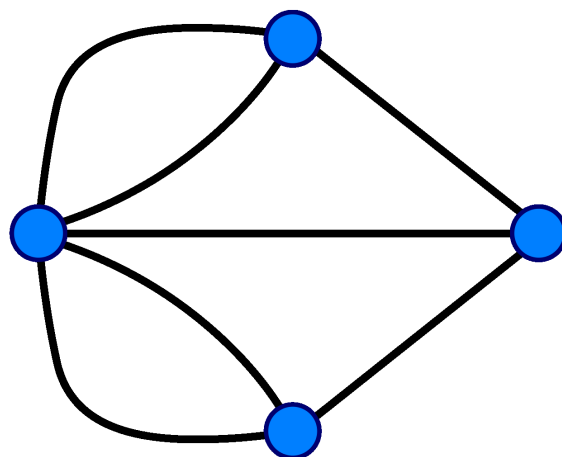
Obrázek 1.2.3: 7 Mostů v Königsbergu - historická mapa



Obrázek 1.2.4: 7 Mostů v Königsbergu - idealizace



Obrázek 1.2.5: 7 Mostů v Königsbergu - idealizace



Obrázek 1.2.6: 7 Mostů v Königsbergu - Eulerův graf

1.2.5 Matematické problémy tisíciletí

Starověkým Řekům by jistě připadala teorie komplexních čísel nebo neeukleidovských geometrií nesrozumitelná a nepotřebná. Nám tak dnes připadá moderní matematika. Ovšem to, co se řeší v matematice nyní, bude třeba časem považováno za běžnou matematiku 22. století.

Problémy tisíciletí (anglicky *Millenium Prize Problems*) je označení pro sedm matematických problémů, které v roce 2000 vyhlásil Clayův matematický institut jako nejdůležitější otevřené problémy soudobé matematiky. Jsou tak jakousi obdobou Hilbertových problémů ze začátku dvacátého století. Na vyřešení každého z nich vypsál institut odměnu jednoho milionu dolarů. Do této chvíle byla vyřešena pouze Poincarého domněnka.

Problém P versus NP je jediný ze 7 problémů tisíciletí, který se týká počítačů. Otázka zní, zda je třída složitosti NP rovna třídě složitosti P. Jinými slovy, zda problémy,

pro něž lze jejich řešení ověřit v polynomiálním čase, lze též v polynomiálním čase vyřešit.

Hodgeova domněnka se týká topologie a tvrdí, že pro projektivní algebraické variety jsou Hodgeovy cykly racionální lineární kombinací algebraických cyklů.

Poincarého domněnka se rovněž týká topologie a jako jediná z problémů tisíciletí byla již vyřešena. Důkaz podal roku 2003 *Grigorij Perelman*; jeho správnost byla potvrzena v srpnu 2006. Domněnka tvrdí, že každý jednoduše souvislý trojrozměrný povrch je ekvivalentní povrchu čtyřrozměrné krychle.

Riemannova hypotéza je jediným dosud nevyřešeným problémem z *Hilbertova seznamu*. Formuloval ji již v roce 1859 *Bernhard Riemann*. Hypotéza spojuje elegantním způsobem matematickou analýzu a teorii čísel a má hluboký význam pro rozložení prvočísel. Tvrdí, že všechny netriviální nulové body Riemannovy funkce *zeta* mají reálnou část rovnou $1/2$. Říká se, že její vyřešení by způsobilo senzaci v teorii kódování.

Yangova-Millsova teorie a hypotéza hmotnostních rozdílů *Yangovy-Millsovy rovnice* popisující chování elementárních částic jsou zobecněnou verzí *Maxwellových rovnic*.

Navierovy-Stokesovy rovnice jsou parciální diferenciální rovnice, které popisují proudění kapalin a plynů. Byly formulovány již v 19. století, dosud však není jasné, zda pro dané počáteční podmínky existuje jejich řešení. Úspěšné vyřešení tohoto problému by například přispělo k porozumění turbulent.

Birchova a Swinnerton-Dyerova domněnka tvrdí, že pro jistý typ rovnic existuje relativně jednoduchý způsob, jak určit, zda má daná rovnice konečný nebo nekonečný počet řešení v racionálních číslech. Pro obecné diofantické rovnice bylo důkazem *Matijasevičovy věty* prokázáno, že nelze dokonce ani určit, zda rovnice má vůbec nějaké řešení.

Nezasvěcená osoba dnes nemá šanci se do některých soudobých problémů matematiky možnost zapojit. Hloubka a abstrakce dosáhly takové úrovně, že už jí může stačit jen špičkový odborník.

Teprve koncem devatenáctého století byla dokončena přesná teorie reálných čísel. Ještě i dnes, navzdory jednoduché představě o reálných číslech jakožto o bodech na přímce, je jejich formální (a vysoce abstraktní) definice postrachem pro vysokoškolské studenty.

V 18. století způsobil v geometrii odborníkům i nematematikům podobné obrovské koncepční potíže objev existence jiných geometrií než té, kterou popsal Eukleides ve své slavné knize *Základy*.

Myšlenka „neeuclidovských geometrií“ se dočkala obecného přijetí až v devatenáctém století. Ke smíru došlo bez ohledu na to, že náš každodenní svět je výhradně eukleidovský.

Volně zpracováno podle *wikipedie* a knihy *Keith Devlin: Problémy pro třetí tisíciletí. Sedm největších nevyřešených otázek matematiky*.

1.2.6 Hilbertovy problémy

Seznam 23 takzvaných *Hilbertových problémů* předložil David Hilbert (1862-1943) v roce 1900 ve své přednášce *Problémy matematiky* na 2. mezinárodním kongresu matematiků v Paříži. Tyto problémy představovaly největší tehdy nevyřešené matematické problémy. Velká část těchto problémů je již dnes vyřešena, přičemž jejich řešení významně ovlivnilo matematiku 20. století.

Seznam problémů je v následující tabulce:

	Stav	Popis	Příbuzné téma
1a.	vyřešeno	Existuje množina, jejíž kardinalita by byla menší než kardinalita množiny reálných čísel, ale větší než kardinalita spočetné nekonečné množiny?	Hypotéza kontinua
1b.	vyřešeno	Je kontinuum reálných čísel dobře uspořádaná množina?	Axiom výběru
2.	vyvráceno	Lze dokázat, že axiomy logiky jsou bezesporné?	Gödelova věta
3.	vyvráceno	Existují dva mnohostěny o stejném objemu, které není možno přeskládat jeden v druhý?	3. Hilbertův problém
4.	v podstatě vyřešeno	Najděte geometrie, které jsou při zachování axiomů přímky a kružnice oslabení axiomu pravého úhlu a odstranění axiomu o rovnoběžkách nejbližší Eukleidovské geometrii.	4. Hilbertův problém
5.	dokázáno	Lze u funkcí definujících spojitou transformační grupu automaticky předpokládat diferencovatelnost?	Lieova grupa
6.	otevřený problém	Axiomatizujte fyziku.	6. Hilbertův problém
7.	částečně dokázáno	Je a^b transcendentní číslo pro každé algebraické číslo $a \neq 0, 1$ a iracionální číslo b ?	Gelfondova věta
8.	otevřený problém	Dokažte Riemannovu hypotézu.	Riemannova hypotéza
9.	částečně vyřešeno	Zobecněte reciproční věty teorie čísel.	Artinova věta
10.	vyvráceno	Existuje algoritmus, který je schopen pro libovolnou diofantickou rovnici určit, má-li řešení?	Matijasevičova věta
11.	částečně vyřešeno	Zobecnění teorie kvadratických forem pro libovolné těleso konečného stupně.	11. Hilbertův problém
12.	v podstatě vyřešeno	Zobecnění Kroneckerovy věty pro obecné algebraické těleso.	Takagiho věta
13.	částečně dokázáno	Dokažte, že obecnou rovnici sedmého stupně nelze vyřešit pomocí kompozice funkcí dvou proměnných.	Arnol'dova věta
...			

Stav	Popis	Příbuzné téma
14. v podstatě vyvráceno	Důkaz konečnosti jistého systému funkcí.	14. Hilbertův problém
15. vyřešeno	Rigorózně podložit algebraickou geometrii, obzvláště Schubertovu výpočetní geometrii.	Schubertova výpočetní geometrie
16. otevřený problém	Studium topologie reálných algebraických křivek a povrchů.	16. Hilbertův problém
17. vyřešeno	Nalézt způsob, jak vyjádřit definitní racionální funkci jako součet čtverců racionálních funkcí.	17. Hilbertův problém
18. vyřešeno	Lze zaplnit prostor pomocí nepravidelného opakování mnohostěnů?	18. Hilbertův problém
19. vyřešeno	Jsou řešení problémů ve variačním počtu vždy algebraická?	19. Hilbertův problém
20. v podstatě vyřešeno	Řešení obecných úloh s okrajovými podmínkami.	Dirichletův problém
21. vyřešeno	Řešení lineárních diferenciálních rovnic při zadané charakteristické grupě.	21. Hilbertův problém
22. vyřešeno	Uniformizace komplexních analytických funkcí pomocí automorfních funkcí.	22. Hilbertův problém
23. v podstatě vyřešeno	Rozšíření metod variačního počtu.	Variační počet

■

Kapitola 2

Algoritmy

Obsah kapitoly

2.1	Co je to algoritmus?	19
2.1.1	Různá pojetí algoritmu	20
2.1.2	Reprezentace a zápis algoritmu	21
2.2	Obecné vlastnosti algoritmů	22
2.3	Způsoby klasifikace algoritmů	23
2.3.1	Dělení algoritmů	24
2.3.2	Zpracování jednotlivých kroků algoritmu	25
2.3.3	Nároky na zdroje (potřeba zdrojů)	25
2.3.4	Třídy složitosti	26
2.4	Časová složitost algoritmů	27
2.4.1	Časová složitost nalezení maxima	27
2.4.2	Časová složitost algoritmu <i>bubblesort</i>	28
2.4.3	Ostatní časové složitosti	29
2.5	Asymptotická složitost algoritmů	30
2.5.1	Asymptotická složitost algoritmu <i>bubblesort</i>	30
2.5.2	Asymptotická složitost algoritmu <i>quicksort</i>	31
2.5.3	Problémy asymptotické složitosti	33

Algoritmy a algoritmizace jsou pojmy, které v současnosti „hýbou světem“. Kdekdo je používá, často v nesmyslných souvislostech, málokdo jim rozumí.

V následujících odstavcích se pokusíme tyto pojmy poněkud lépe „pojmenovat“, uvést základní vlastnosti a na příkladech ukázat vybrané sledované vlastnosti některých algoritmů. Bohužel není možné se této problematice věnovat v její plné šíři, tolik „prostoru“ v tomto sešitu k dispozici není.

2.1 Co je to algoritmus?

Problematika algoritmů a algoritmizace je dnes natolik široká¹, že ji není možné postihnout jednou vědní specializací. Dokonce ani jedním vědním oborem. Tématu se

¹hluboká taktéž

„nějak“ dotýkají vědní obory, jako například:

- matematika (teorie grafů, numerická matematika, problémy typu P-NP,...)
- teoretická informatika (překladače, jazyky, datové struktury, kybernetika, ...)
- aplikované vědy (databáze, operační systémy, grafické algoritmy, programování, řadičů a vyhledávací algoritmy, ...)

Budeme chtít různorodé algoritmy nějak posuzovat, hledat lepší, rychlejší, efektivnější²,... K tomu potřebujeme nějaká kritéria, podle kterých je budeme „nějak měřit“, posuzovat nějaké (vybrané) vlastnosti.

2.1.1 Různá pojetí algoritmu

V běžné řeči a v různých učebnicích se lze setkat s různě přísným pojetím pojmu algoritmus. Někdo požaduje splnění více z uvedených vlastností, někdo méně.

Základní myšlenka je nicméně shodná. Algoritmus je lidsky (a tedy nepřesně) řečeno strojově proveditelný, spolehlivý a užitečný pracovní postup.

Různé varianty definice *algoritmu*, které jsme v (konečném čase) našli:

1. Přesný návod, postup, souhrn elementárních kroků, činností vedoucích k dosažení požadovaného cíle.
2. Algoritmus je pracovní postup, který splňuje tyto povinné vlastnosti:
 - (a) **Rezultativnost**. To znamená, že vždy vydá nějaký výsledek.
 - (b) **Finitnost** (konečnost). To znamená, že někdy (v konečném čase) skončí. Jinými slovy, skončí po konečném počtu provedených kroků.
 - (c) **Elementárnost** (jednoduchost) popisu. Algoritmus je popsán konečným počtem základních instrukcí. Tedy takových, o kterých je jasné, jak se provedou (a tedy neumožňují žádný osobitý výklad některého vykonavatele).
 - (d) **Determinovanost** (jednoznačnost). Postup práce je jasně daný a vždy závisí pouze na popisu algoritmu a na vstupu (ať už jde o vejce, nebo o informace, tedy nějaká čísla). Na průběh algoritmu nemá žádný vliv.
3. Algoritmus numerické metody je jasná a jednoznačná specifikace (popis) konečné posloupnosti operací, aritmetických a logických, jejichž prostřednictvím se m – tici čísel z určité množiny vstupních dat jednoznačně přiřazuje n – tice výsledků.
4. Numerický algoritmus (=numerický proces) neobsahuje logické operace.
5. Algoritmus je přesný návod či postup, kterým lze vyřešit daný typ úlohy. Pojem algoritmu se nejčastěji objevuje při programování, kdy se jím myslí teoretický princip řešení problému (oproti přesnému zápisu v konkrétním programovacím jazyce).

²všimneme si, že uvedené pojmy mají zatím poněkud nejasný, neurčitý obsah

6. Algoritmus je „posloupnost konečného počtu elementárních kroků vedoucí k vyřešení úlohy“. Jinými slovy, algoritmus je přesný postup, jakým lze vyřešit daný úkol (např. zjistit součin dvou čísel). Kvalitu algoritmu pak určuje nejmenší počet kroků, který je k dosažení cíle potřeba.

Blok 2.1.1: Možná pojetí algoritmu

Jako příklad algoritmu se často uvádí kuchařský recept, coby popis pracovního postupu vedoucího k dosažení cíle (zhotovení pokrmu).

Je dobré si uvědomit, že termín *algoritmus* má mnohem obecnější platnost. Postup, kdy se přesouváme z bodu *A* do bodu *B* také vyhovuje kritériím algoritmu. Přecházíme-li vozovku, je dobré se rozhlédnout, nejdříve vlevo, poté vpravo. Jede-li z patřičného směru nějaké vozidlo, pak. . .

Lze nahlédnout, že uvedený příklad může být součástí většího celku, například nalézt vhodnou trasu z bodu *A* do bodu *B*. Ze zkušenosti víme, že takových tras může být velké množství. Pak přijde na řadu „dílčí“ algoritmus přesunu z bodu *A* do bodu *B*. . .

2.1.2 Reprezentace a zápis algoritmu

Zde je vysvětleno několik základních způsobů, jak můžeme zapsat algoritmy. Správně zapsat algoritmus je důležité především pro jeho další využití, případně úpravu.

Slovní popis – slovní popis v přirozeném jazyce. Vysvětlení krok po kroku, co algoritmus v dané fázi provádí a jak bude reagovat.

Vývojový diagram – grafické znázornění algoritmu, řešení úlohy jednotnými značkami a zkratkami.

Strukturogram – Úspornější znázornění algoritmu, kombinace grafického a textového popisu

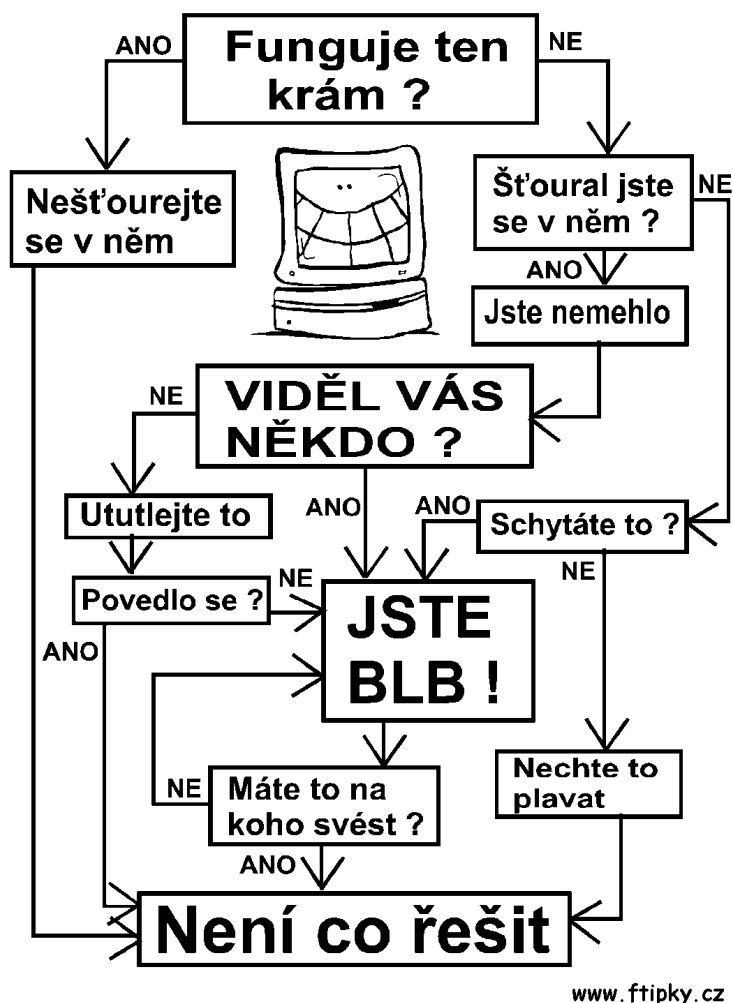
Datově orientovaný diagram HIPO – (z angl. Hierarchy plus Input- Process- Output), je grafickým vyjádřením funkčního členění problému, struktury dat a postupu řešení problému při různém stupni podrobnosti.

Rozhodovací tabulka – používá se zejména při složitějších větveních, tabulka rozdělená do kvadrantů.

Stavové diagramy – grafické znázornění algoritmu. Podobné vývojovým diagramům, vychází však z odlišné filozofie přístupu k řešení algoritmu.

Existují další, většinou však úzce specializované popisy činností. Často se používají k popisu pouze k jedné, (nebo několika málo) konkrétní činnosti. Také mohou případně popisovat specifické datové struktury. Typický příklad jsou tzv. „UML diagramy“.

TECHNOLOGICKÉ SCHÉMA ŘEŠENÍ PROBLÉMU



Obrázek 2.1.1: Příklad algoritmu (stavového diagramu)

2.2 Obecné vlastnosti algoritmů

Postupem doby došlo k obecné shodě o některých základních vlastnostech algoritmů a způsobech (metodách) stanovení kvantifikačních parametrů. Algoritmy by měly mít obecné vlastnosti:

1. **konečnost** (finitnost)

Každý algoritmus by měl (musí) skončit v konečném počtu kroků. Počet kroků může být libovolně velký (v závislosti na rozsahu vstupních údajů, příp. jejich hodnotách), ale pro každý jednotlivý vstup musí být konečný. Postupy, které tuto podmínku nesplňují, se obvykle nazývají *výpočetní metody*. Příkladem nekonečné výpočetní metody je reaktivní proces, který průběžně reaguje s okolním prostředím, například operační systém. Někteří autoři však mezi algoritmy zahrnují i takovéto postupy.

2. **Obecnost** (univerzálnost)

Algoritmus nemá řešit jednu konkrétní úlohu (např. „vypočítej rovnici $y = 2 \cdot x + 4$ “, ale obecnou třídu problémů daného typu, (například „vypočítat rovnici ve tvaru $y = k \cdot x + q$ “. Musí mít širokou množinu možných vstupů (validních i nevalidních) včetně vhodné reakce.

3. **Determinovanost**

Každý krok algoritmu musí být jednoznačně a přesně definován. V každé situaci musí být naprosto zřejmé, co a jak se má provést, jak má provádění algoritmu pokračovat. Pro stejná vstupní data musíme dostat vždy stejné výsledky. Protože běžný (hovorový) jazyk obvykle nebývá zcela jednoznačný, neposkytuje naprostou přesnost, byly pro zápis algoritmů navrženy zvláštní (programovací) jazyky, které tento požadavek splňují. V nich má každý příkaz jasně definovaný jednoznačný význam. Vyjádření výpočetní metody v programovacím jazyce se nazývá program. Některé algoritmy ale determinované nejsou, hovoříme potom o *nedeterministických algoritmech*.

4. **Výstup** (resultativnost)

Algoritmus má alespoň jeden výstup, veličinu, která je v požadovaném vztahu ke vstupním (zadaným) datům. Tvoří tak odpověď na problém, který algoritmus řeší (algoritmus vede od zpracování hodnot k výstupu).

5. **Elementárnost**

Algoritmus se skládá z konečného počtu relativně jednoduchých (elementárních) kroků. Měl by i po konečném počtu kroků skončit. Je dobré si uvědomit, že zapsaný konečný počet kroků algoritmu v nějakém programovacím jazyce ještě nemusí znamenat konečný počet *vykonaných kroků*.

Lze nahlédnout, že uvedená kritéria jsou spíše kvalitativní, než kvantitativní. Nás budou především zajímat takové algoritmy, které jsou v nějakém smyslu *kvalitní*. Takové algoritmy hodnotíme podle různých kritérií, která nějak splňují. Například měříme počet kroků potřebných pro běh algoritmu, jednoduchost, efektivitu či „eleganci“.

Problematikou efektivity algoritmů, metodami, jak z několika algoritmů řešících konkrétní problém vybrat ten vhodný, se zabývají odvětví informatiky nazývané algoritmická analýza a teorie složitosti. Některé metody si v následujících odstavcích ukážeme.

2.3 Způsoby klasifikace algoritmů

V odstavci (2.1) jsme se pokusili naznačit, že pod pojem *algoritmus* lze v podstatě „nějak“ zahrnout jakoukoliv činnost, proces ve společnosti, příp. v přírodě. Je tedy zřejmé, že neexistuje jedno univerzální hodnotící kritérium, podle kterého by bylo možné všechny algoritmy jednotně, univerzálně „změřit“.

Každá oblast použití si tak vyvinula (stanovila) své postupy hodnocení „kvality“, případně „efektivity“ použitých algoritmů.

Postupem doby se tak odborná veřejnost shodla na určitých metodologiích, kterak hodnotit algoritmy. Uvedeme si pouze některá, (asi) nejčastěji používaná kritéria. Ostatní,

méně často používaná, případně úzce (specificky) zaměřená kritéria si laskavý čtenář jistě (s pomocí „Velkého Bratra“) vygooglí sám.

2.3.1 Dělení algoritmů

Iterativní – algoritmus opakuje určité části (skupinu instrukcí). Každé opakování může probíhat s jinými daty, myšleno, jako „vstupní data“ pro blok instrukcí se použije „výsledek“ z předchozího iteračního kroku.

Typický představitel iterativního způsobu práce může být elementární krok známého algoritmu *bubblesort*:

je-li: $prvek_i > prvek_{i+1} \rightarrow$ zaměň obsah p_i a p_{i+1} , pro: $i = 0, 1, 2, \dots, n-1$.

Rekurzivní – algoritmus „volá“ sám sebe s novou sadou parametrů. Často jsou k těmto účelům využity zásobníkové datové struktury poskytované (obvykle) operačním systémem. Typický představitel je definice výpočtu faktoriálu:

$$n! = n \cdot (n-1)!$$

případně častá implementace řadicího algoritmu *Quick sort*.

Rekurzivní algoritmus opakuje kód prostřednictvím volání sebe sama (obvykle na podproblémech menší velikosti). Každý rekurzivní algoritmus lze převést do iterativní podoby. Samotný převod často řeší automaticky kompilátor nebo virtuální stroj daného programovacího jazyka.

Výhoda rekurzivních algoritmů je v jejich snadno čitelném a kompaktním zápisu. Nevýhodou je spotřeba dodatečných systémových prostředků (zdrojů) pro udržení jednotlivých rekurzivních volání.

Hladové – pracují se k řešení pomocí jednotlivých rozhodnutí, která jakmile jsou učiněna, už nejsou dále revidována

Algoritmy typu rozděl a panuj – rozdělí problém na menší podproblémy a dílčí řešení poté slučují. Typický představitel této třídy algoritmů je algoritmus *Quick sort*.

Algoritmy dynamického programování – řeší problémy od nejjednodušších po složitější, přičemž využívají výsledky již vyřešených jednodušších problémů.

Pravděpodobnostní algoritmy – některá rozhodnutí provádějí náhodně.

Genetické algoritmy – napodobují biologické evoluční procesy.

Heuristické algoritmy – nehledají přesná řešení, pouze nějaká vhodná „přiblížení“.

(Ne)deterministické algoritmy – jsou takové algoritmy, které mají v každém svém kroku právě jednu možnost, jak pokračovat. Nedeterministický jich má více. Příkladem může být deterministický a nedeterministický automat.

2.3.2 Zpracování jednotlivých kroků algoritmu

sériové algoritmy – jednotlivé kroky algoritmu jsou prováděny jeden za druhým, *serializovaně*, následující krok se neprovede dříve, dokud se nedokončí předchozí krok. V podstatě (dodávna) jediný možný způsob práce na jednoprosesorových systémech.

Řada algoritmů je navržena tak, že pracují *serializovaně*, nelze tudíž využít možností současných výpočetních systémů. Takové algoritmy je nutné přeformulovat, je-li to možné.

paralelní algoritmy – jednotlivé kroky algoritmu (bloky kroků) se zpracovávají *zároveň* v čase na více procesorech. Tento způsob práce klade extrémní nároky na správné sestavení (formulování) algoritmu z pohledu zachování *konzistence dat*. Používá se obvykle na výpočetních systémech *těsně vázaných* (více výpočetních jader, společná paměť) a více, či méně složité *synchronizační metody*.

distribuované algoritmy – je varianta paralelního počítání s tím, že „vazba“ výpočetních systémů je poměrně volná (hovoříme o *výpočetních uzlech*). Často výpočet probíhá tak, že určitá centrální autorita, *supervizor* rozdělí data na dílčí bloky a ty poté „přidělí“ jednotlivým výpočetním uzlům. Každý výpočetní uzel pak disponuje vlastním výpočetním jádrem, vlastní pamětí a (nežádá) i vlastním operačním systémem. Typický představitel distribuovaného počítání je projekt *Boinc* (dříve *SETI@home*). Nároky na „vhodné sestavení“ algoritmu, či spíše jeho vhodném rozdělení, distribuci, budou podobné jako v případně paralelních algoritmů. Podobné závěry lze odvodit i pro „počítaná data“.

2.3.3 Nároky na zdroje (potřeba zdrojů)

Máme za úkol vykonat, zpracovat jistý proces. Tento proces je charakterizován určitým pracovním postupem, algoritmem. Abychom mohli úspěšně (často i neúspěšně) provést takový proces, potřebujeme k tomu určité podmínky. Například pro uvaření (možná) dobrého oběda potřebujeme

- předepsané suroviny
- pracovní postup
- kuchyň s nádobím, ap.

Pro zpracování určitého výpočetního postupu také potřebujeme určité podmínky.

- vhodně vybavený počítač
- vytvořený program
- čas na výpočet

Je patrné, že zakoupení předepsaných surovin pro uvaření oběda stojí určitou sumu peněz. Jaké suroviny a v jakém množství nakoupíme je závislé na počtu porcí, které máme uvařit a na zpracovávaném receptu.

Pořízení počítače stojí určitou sumu peněz. Podle výpočetních nároků použitého algoritmu rozhodneme, kolik a jakého technického vybavení zaplatíme. Například kolik operační paměti potřebujeme, jak výkonný (a drahý) mikroprocesor potřebujeme, ...

Potřeba určité velikosti operační paměti, příp. výkonu procesoru nazýváme termínem *potřeba zdrojů*. Bez ní výpočetní proces neproběhne, využíváme ji, „čerpáme“ její schopnost „pamatovat si“. Zdroje jsou obecně splněné předpoklady pro provedení daného algoritmu.

Kolik a jak velké množství *zdrojů* náš algoritmus potřebuje významně ovlivní nároky na pořízení potřebných prostředků. Proto je nutné znát nároky jednotlivých algoritmů na jednotlivé zdroje.

Asymptotická složitost algoritmu – určitým způsobem vyjadřuje množství času potřebné na provedení algoritmu v závislosti na velikosti zpracovávaných dat. V současnosti je *asymptotická složitost* považována za jeden z nejdůležitějších (sledovaný) parametr popisující vlastnosti algoritmů. Budeme se jí věnovat podrobněji v následujících odstavcích.

Prostorová složitost algoritmu – určitým způsobem vyjadřuje nároky na paměťové kapacity výpočetního systému. Rozlišují se nároky na operační paměť a nároky na diskové kapacity, případně „ostatní“ paměťové kapacity. S současností není prostorová složitost přikládán tak velký význam (dokoupit další operační paměť je relativně snadné a poměrně laciné).

Doba tvorby programu (zdrojového kódu) – reprezentuje časové (a obvykle i finanční) náklady na vytvoření (naprogramování) výpočetního systému. Může nastat situace, že výpočet neklade příliš vysoké nároky na množství dat, případně extrémně krátké časové reakce. Také se může stát, že bude výpočet proveden pouze jednou.

V těchto případech může být výhodnější naprogramovat řešení úlohy rychle, i když výpočetně nepříliš efektivně. Vždy je ovšem nutné takové okolnosti důkladně zvážit.

2.3.4 Třídy složitosti

Třída složitosti je určitou abstrakcí nad různorodou, často nepřehlednou klasifikací různých algoritmů. Stanovuje obtížnost rozhodnutelnosti daného problému na Turingově stroji.

Třída P – polynomiální třída, obsahuje problémy rozhodnutelné v polynomiálním čase. Například:

- Má daný graf kostru o velikosti maximálně k ?
- Existuje v daném acyklickém grafu mezi uzly a a b cesta, jejíž délka je nejvýše k ?

Třída NP – nepolynomiální třída, obsahuje problémy, které jsou rozhodnutelné pomocí nedeterministického Turingova stroje v polynomiálním čase, tzn. jsme schopni ověřit jejich řešení v polynomiálním čase. Například:

- Lze daný graf obarvit maximálně k barvami?

- Existuje v daném grafu hamiltonovská kružnice?
- Existuje v daném grafu klika o alespoň k vrcholech?

2.4 Časová složitost algoritmů

V současnosti lze registrovat explozivní vývoj výkonů (mikro)procesorů, včetně všech různých výpočetních konstrukcí na nich založených. To nabízí, přímo vnucuje, přirozené řešení problému „neúměrně dlouhého výpočtu“. Prostě koupíme rychlejší počítač. První problém tedy nastal, jak vlastně „ohodnotit“ výkon výpočetního systému, co je „rychlé“ a co „pomalé“.

Druhý problém je ještě závažnější. Neznalost závislosti spotřeby výpočetního času na množství zpracovávaných dat může vést k neřešitelným problémům ve vztahu dodavatel-odběratel. Výpočetní systém může velmi uspokojivě běžet pro určité množství dat. Poté dosáhnou data kritické velikosti a systém se začne dramaticky zpomalovat. Proč?

Řešení v rozhodnutí koupit výkonnější systém nemusí problém vyřešit. Otázka totiž zůstává. *Jak moc* výkonnější by měl být nový systém. Problém dále vyhrocuje nelineární závislost ceny výpočetního systému na jeho výkonu.

Bylo by dobré mít k dispozici nějaký *posuzovací aparát*, pomocí kterého bychom dokázali stanovit kritické (časové) charakteristiky systému (algoritmu).

Jedním z nich je *stanovení časové složitosti* algoritmu.

2.4.1 Časová složitost nalezení maxima

Ukažme si na jednoduchém elementárním příkladu princip metody.

Úloha:

V datovém poli o n prvcích najděte maximální prvek

Řešení:

I velmi začínajícímu programátorovi je jasné, jak bude probíhat výpočet. Je patrné, že elementární *krok výpočtu* je právě test, kdy kontrolujeme, zda i – tý prvek není nové maximum. Pro N prvků v poli musíme tedy provést N porovnání.

Označme T_L dobu nutnou pro provedení jednoho elementárního kroku. Lze nahlédnout, že pro N_1 prvků provedeme N_1 porovnání a spotřebujeme $T_1 = T_L \cdot N_1$ času. Pro $N_2 = 10 \cdot N_1$ prvků provedeme $N_2 = 10 \cdot N_1$ porovnání a spotřebujeme $T_2 = T_L \cdot 10 \cdot N_1$ času. Desetinásobné množství vstupní dat povede na desetinásobné množství spotřeby času:³

$$\frac{N_2}{N_1} \approx 10; \quad \frac{T_2}{T_1} \approx \frac{T_L \cdot 10N_1}{T_L \cdot N_1} \approx 10 \approx C_1$$

Je patrné, že spotřeba času výpočtu je lineární funkcí počtu vstupních dat. Hovoříme o *lineární časové složitosti*.

³formální odvození časové složitosti provedeme později. . .

2.4.2 Časová složitost algoritmu *bubblesort*

Ukažme nyní na příkladu algoritmu *bubblesort* jiný typ časové složitosti. Jde o jeden z nejjednodušších řadicích algoritmu.

Úloha:

Datové pole o N prvcích seřídíte vzestupně pomocí algoritmu *bubblesort*.

Řešení:

Ve stručnosti připomeňme v tabulce 2.4.1 funkci algoritmu *bubblesort*.

Funkce algoritmu *bubblesort*:

- a) porovnej hodnotu prvku p_i a p_{i+1} pole
- b) platí-li: $p_i > p_{i+1}$, $\rightarrow$ zaměň obsah prvků p_i a p_{i+1} .
- c) jinak nedělej nic
- d) opakuj body a až c pro všechny prvky pole
- e) opakuj bod d dokud bude splněna (alespoň jednou) podmínka b

Tabulka 2.4.1: Algoritmus *bubblesort*

Zajímá nás, kolikrát musíme provést jednotlivé kroky algoritmu (body a až c tabulky 2.4.1). Předpokládejme nyní, že prvek p_N obsahuje hodnotu *minima* pole.

Je zřejmé, že provedením bodu d:

- se prvek p_N přesune na místo prvku p_{N-1} .
- provedeme právě $N - 1$ elementárních kroků algoritmu (body a až c).

Má-li se (původní) prvek p_N postupně dostat na pozici prvku p_1 je patrné, že bod d algoritmu musíme opakovat právě $N - 1$.⁴

Celkový počet kroků algoritmu tak bude:

$$(N - 1) \cdot (N - 1) = N^2 - 2N + 1$$

Podobně, jako v příkladu ?? nás nyní bude zajímat spotřeba času pro různě velká vstupní data.

Pro vykonání jednoho elementárního kroku algoritmu potřebujeme T_b času. Jsou to kroky a až c. Je také zřejmé, že krok b se nemusí provést pokaždé, záleží na konkrétním obsahu porovnávané dvojice.

Pro N_1 vstupních dat bude potřeba času:

$$T_1 = T_b \cdot (N_1^2 - 2N_1 + 1)$$

Podobně pro $N_2 = 10 \cdot N_1$ vstupních dat spotřebujeme množství času:

$$T_2 = T_b \cdot (N_2^2 - 2N_2 + 1) = T_b \cdot ((10 \cdot N_1)^2 - (2 \cdot 10 \cdot N_1) + 1)$$

⁴nyní neuvažujeme možné optimalizační strategie

Tudíž:

$$\frac{N_2}{N_1} \approx 10; \quad \frac{T_2}{T_1} \approx \frac{T_b \cdot 100N_1^2 - 20N_1 + 1}{T_b \cdot N_1^2 - 2N_1 + 1} \approx 100$$

Byť se zatím jedná spíše o intuitivní odhad, lze konstatovat, že časová závislost algoritmu *bubblesort* je popsána kvadratickým polynomem, říkáme, že je *kvadratická*.

2.4.3 Ostatní časové složitosti

Protože algoritmů je velmi mnoho různých druhů řešící rozmanité úlohy (všeho druhu), lze očekávat i rozmanité časové složitosti. V tabulce (2.4.2) je přehled nejčastějších časových složitostí, které mnohou algoritmy nabývají:

	$n = 10$	$n = 100$	$n = 1\,000$	$n = 10\,000$	$n = 100\,000$
$\log n$	2.3	4.6	6.9	9.2	11.5
$\sqrt{n}$	3.2	10	31.6	100	316.2
n	10	100	1000	10000	100000
$n \log n$	23.0	46.0	69.1	92.1	115.1
n^2	100	10^4	10^6	10^8	10^{10}
n^3	1000	10^6	10^9	10^{12}	10^{15}
2^n	1024	$1.26 \cdot 10^{30}$	$1.07 \cdot 10^{301}$	$\approx \infty$	$\approx \infty$
$n!$	$3.63 \cdot 10^6$	$9.36 \cdot 10^{157}$	$\approx \infty$	$\approx \infty$	$\approx \infty$

Tabulka 2.4.2: Časté časové složitosti a počty operací

Zkusme nyní odhadnout kolik času by potřeboval nějaký výpočetní systém k dokončení úlohy, budou-li čísla v tabulce představovat počty operací k dokončení výpočtu.

Optimisticky předpokládáme, že máme k dispozici výpočetní systém, který je schopen provést 10^{10} operací/sekundu. Je patrné, že máme výpočetní systém extrémně výkonný a také extrémně drahý.

	$n = 10$	$n = 100$	$n = 1\,000$	$n = 10\,000$	$n = 100\,000$
$\log n$	0.23ns	0.46ns	0.96ns	0.92ns	1.15ns
$\sqrt{n}$	0.32ns	1ns	3.16ns	10ns	31.62ns
n	1ns	10ns	100ns	1μs	10μs
n^2	10ns	1μs	100μs	10ms	1s
n^3	100ns	100μs	0.1s	100s	27.8hod
2^n	102.4ns	$8.59 \cdot 10^{14} \text{let!}$	$3.39 \cdot 10^{283} \text{let!}$	$\approx \infty$	$\approx \infty$
$n!$	363μs	$2.96 \cdot 10^{140} \text{let!}$	$\approx \infty$	$\approx \infty$	$\approx \infty$

Tabulka 2.4.3: Spotřeba času pro různé časové složitosti

Pechlivým prostudováním tabulky 2.4.3 zjistíme, že pro běžnou výpočetářskou praxi lze⁵ rozumně akceptovat časovou složitost stupně n^2 , maximálně (ve speciálních případech) složitost stupně n^3 .

Lze také konstatovat, že pro časové složitosti *exponenciálního*, případně *faktoriálního* stupně se s největší pravděpodobností *nikdy* nepodaří vyrobit použitelný výpočetní systém pro data určité, poměrně malé velikosti.

Možná pomůže k vytvoření určité časové představy srovnání, stáří vesmíru se odhaduje na „pouhých“ $4.35 \cdot 10^{17} \text{sec}$...

⁵někdy se skřípěním zubů...

2.5 Asymptotická složitost algoritmů

Analýzou algoritmu *bubblesort* jsme dospěli k závěru, že jeho časovou složitost je možné vyjádřit formulí:

$$\frac{N_2}{N_1} \approx 10; \quad \frac{T_2}{T_1} \approx \frac{T_b \cdot 100N_1^2 - 20N_1 + 1}{T_b \cdot N_1^2 - 2N_1 + 1} \approx 100$$

Pro praktické potřeby porovnávání vlastností různých algoritmů je tato, poměrně složitě zapsaná, formule obtížně použitelná. Lze odhadnout, že pro jiné algoritmy může být formule vyjadřující jeho časovou složitost podstatně složitější. Porovnání „kvality“ algoritmů z hlediska jejich časových nároků by tak bylo poměrně složité a obtížné. Také nás spíše zajímá chování algoritmů pro *opravdu velká data*.

2.5.1 Asymptotická složitost algoritmu *bubblesort*

Jak vyřešit tento problém? Využijeme faktu, že pro *opravdu velká data* je „časový příspěvek“ nejvyššího řádu (obvykle polynomu) ve formuli k celkové spotřebě času dominantní. Ostatní členy (obvykle) polynomu ve formuli tak „přispívají“ k celkové spotřebě času podstatně menší měrou. Rozhodneme tedy, že jejich „příspěvek“ je zanedbatelný. A přesně to uděláme, zanedbáme ho.

Ukažme si tento postup na příkladu algoritmu *bubblesort*. Mějme množství vstupních dat N_1 . Pro jejich zpracování potřebujeme T_1 času. Pro jiné množství vstupních dat $N_2 = 1\,000 \cdot N_1$ bude potřeba T_2 času. Poměříme-li tato čísla, získáme:

$$\frac{N_2}{N_1} = \frac{1000 \cdot N_1}{N_1} \approx 1000; \quad \frac{T_2}{T_1} \approx \frac{T_b \cdot 1\,000\,000N_1^2 - 2\,000N_1 + 1}{T_b \cdot N_1^2 - 2N_1 + 1} \approx 1\,000\,000$$

Lze si všimnout, že pro poměr množství vstupních dat 1 : 10 je příspěvek ostatních členů polynomu cca 20%. Pro poměr množství vstupních dat 1 : 1000 je příspěvek ostatních členů polynomu cca 0.2%. Pro *opravdu velká data* se tak bude limitně blížit nule.

Než se dostaneme k formální definici *asymptotické časové složitosti*, prozkoumejme ještě jeden aspekt problému spotřeby času. Všímavý čtenář již pravděpodobně zpozoroval, že odhad času pro zpracování algoritmu *bubblesort* je *nejhorším časovým odhadem*. Hůře⁶ výpočet dopadnout nemůže. Co by mohlo všímavého čtenáře vést k tomuto závěru?

Vraťme se ještě jednou k podrobnostem funkce algoritmu *bubblesort*. Víme, že elementární operace algoritmu jsou body **a** až **c** v tabulce 2.4.1. Dobu, kterou tyto operace potřebují ke svému provedení jsme označili T_b . I programátor-začátečník ví, že musí použít nejméně *dvě* různé skupiny instrukcí zvoleného jazyka.

- instrukci pro test,
- instrukce pro postupnou záměnu obsahu proměnných

⁶z hlediska spotřeby času

Použití instrukcí pro záměnu obsahu proměnných bude závislé na konkrétním obsahu tříděných dat. Mohou se provést, ale nemusí. Pokud se neprovedou, nespotřebují procesorový čas. Jaká tedy bude časová závislost algoritmu *bubblesort*, budeme-li respektovat nová zjištění?

Počet elementárních operací typu „test“ bude $n^2 - 2n + 1$. Plyne to z charakteru funkce algoritmu *bubblesort*. Jak postihnout skutečnost, že někdy se záměnné operace provedou, někdy ne. Ukazuje se, že vhodný popis, který dostatečně přesně respektuje dané vlastnosti je:

$$T_1 = T_e \cdot c_1 \cdot (n_1^2 - 2n_1 + 1)$$

pro počet vstupních dat n_1 a „určitou skladbu“ vstupních dat. Pro jinou kombinaci hodnot vstupních dat zjistíme:

$$T_2 = T_e \cdot c_2 \cdot (n_1^2 - 2n_1 + 1)$$

kde $c_1 > 1 ; c_2 > 1$. Charakter, skladbu vstupních dat tedy obecně postihneme nějakou multiplikativní konstantou:

$$c_1 > c > c_2 > 1.$$

kde c_1 je multiplikativní konstanta pro nejhorší časový odhad, c_2 je multiplikativní konstanta pro nejlepší časový odhad.

Multiplikativní konstanta tak pouze zohlední konkrétní dobu výpočtu pro konkrétní skladbu vstupních dat a konkrétní algoritmus.

Budeme-li předpokládat, že se záměna obsahu sousedních prvků provede *vždy*, získáme právě náš *nejhorší časový odhad*.

Odhadněte časovou konstantu c_2 pro algoritmus *bubblesort*

Pozor, je to chyták !

Příklad 2.5.1: Odhad časové konstanty c_2 algoritmu *bubblesort*

2.5.2 Asymptotická složitost algoritmu *quicksort*

V odstavci 2.5.1 jsme zjistili, že stanovená (odhadnutá) časová složitost algoritmu *bubblesort* je právě *nejhorší možná*.

Pak by ale mohl existovat takový odhad časové složitosti, který by mohl být nejlepší možný, pokud analyzovaný algoritmus touto vlastností disponuje. Ukažme si na příkladu algoritmu *quicksort* podobnou vlastnost.

Poznamenejme, že algoritmus *quicksort* lze charakterizovat jako algoritmus typu *rozděl a panuj*. Pracuje *rekurzivně* tak, že rozdělí datové pole na menší úseky podle určitého pravidla. Poté je seřídí, bližší popis viz definice 2.5.2, příp. definice 2.5.3.

Funkce algoritmu *Quicksort*(l, p):

Vstup: $l, p, \text{pole}[\cdot]$
 pokud $l < p$
 $\text{pivot} \leftarrow \text{pole}[(l+p)/2]$
 $q \leftarrow \text{rozdel}(l, p, \text{pivot})$
 $\text{Quicksort}(l, q)$
 $\text{Quicksort}(q+1, p)$
 Výstup: Setříděné pole $p[\cdot]$

Blok 2.5.2: Funkce algoritmu *quicksort*

Funkce *rozdel* rozdělí pole dat na dvě části. Levá polovina obsahuje prvky, které jsou menší, nebo rovny pivotu:

$$\text{pole}[k] \leq \text{pivot} ; \quad k = l, l+1, l+2, \dots, q;$$

Pravá polovina pole obsahuje prvky, které jsou větší než pivot:

$$\text{pole}[k] > \text{pivot} ; \quad k = q+1, q+2, q+3, \dots, p.$$

Hodnota indexu q odpovídá hranici mezi levou a pravou polovinou pole.

Činnost funkce *rozdel*(l, p, pivot):

Vstup: l, p, pivot
 opakuj dokud platí $l < p$
 opakuj dokud platí $\text{pole}[l] < \text{pivot}$
 $l \leftarrow l+1$
 opakuj dokud platí $\text{pole}[p] > \text{pivot}$
 $p \leftarrow p-1$
 zaměň(l, p)
 Výstup: vrať index p

Blok 2.5.3: Činnost funkce *rozdel*

Máme-li štěstí (v ideálním případě), podaří se postupně vybírat jednotlivé pivoty tak, že každý úsek bude rozdělen právě na dvě stejné poloviny. Získáme tak:

nultá hladina rekurze	→ 1 pole o n prvcích
první hladina rekurze	→ 2 pole, každé o $\frac{n}{2}$ prvcích
druhá hladina rekurze	→ 4 pole, každé o $\frac{n}{4}$ prvcích
třetí hladina rekurze	→ 8 polí, každé o $\frac{n}{8}$ prvcích
...	

Tabulka 2.5.4: Algoritmus *quicksort* - dělení pole na useky

Lze vysledovat, že časová složitost pro jednotlivé úseky je:

$$\mathcal{O}(n + 2\frac{n}{2} + 4\frac{n}{4} + 8\frac{n}{8} + \dots + 2^{\log n} \cdot 1) = \mathcal{O}(n \cdot \log n).$$

Což je asymptotická časová složitost, která je „velmi příjemná“. Odtud má také algoritmus své jméno. . .

Co se stane, pokud nemáme takové štěstí, ale naopak, nehoráznou smůlu a „podaří“ se nám postupně vybírat jednotlivé pivoty tak, že jejich hodnota vždy bude nejmenší (nebo největší) číslo pro dané pole?

V takovém případě je z funkce algoritmu zřejmé, že jednotlivá dělení pole na části budou extrémně nesymetrická. Vždy „vzniknou“ pole o velikosti 1 prvek a zbytek pole. Z funkce algoritmu také plyne, že v takovém případě bude počet dělení n (přesně $n - 2$). Potom „příjemná časová složitost“ přejde na tvar:

$$\mathcal{O}(n \cdot \log n) \rightarrow \mathcal{O}(n \cdot n) = \mathcal{O}(n^2).$$

Což je časová složitost, která je podstatně méně výhodná. Pro *opravdu velká data* by pak mohl být rozdíl spotřeby času značný. Tato nepříjemná vlastnost tak „diskvalifikuje“ tento algoritmus v situacích, kdy je nutné garantovat dobu odezvy a kvadratická časová složitost je nevyhovující. . .

2.5.3 Problémy asymptotické složitosti

V odstavci 2.5 jsme naznačili závislost spotřeby strojového času, doby výpočtu na množství vstupních dat.

Na daný problém lze nahlížet i „z druhé strany“. Existuje řada situací (požadavků), kdy na provedení určitého (zvoleného) výpočtu mám jen omezené množství času⁷.

Proto je důležité umět rozhodnout, zda výpočet skončí v požadovaném čase, případně jak „to“ dopadne pro data n – násobné velikosti.

Nastal tak čas formálně definovat *časovou složitost*.

Říkáme:

$$f(n) = \mathcal{O}(g(n)) \text{ právě tehdy } \exists c > 0 ; n > n_0 : f(n) \leq c \cdot g(n).$$

ve významu:

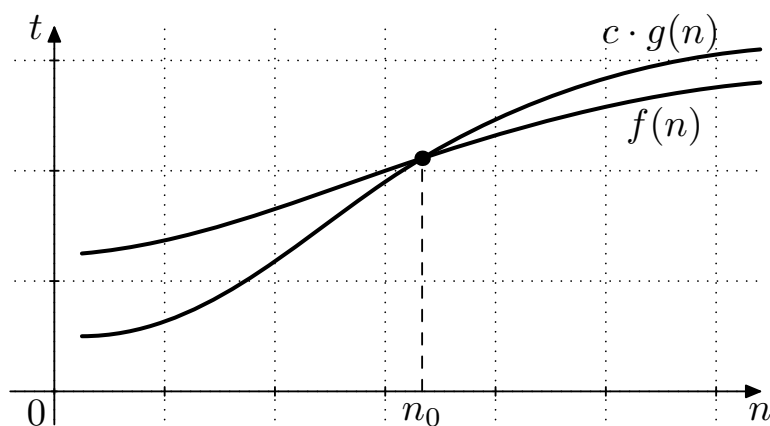
funkce $f(n)$ neroste řádově rychleji, než funkce $g(n)$.

viz obrázek 2.5.2.

Blok 2.5.4: Definice časové složitosti

Důležité je, že se *nezmění* charakter časové složitosti algoritmu.

⁷Co je omezené množství času je velmi závislé na okolnostech. Někdy může být „dost“ i jedna milisekunda, jindy může být jeden den málo. . .



Obrázek 2.5.2: Definice časové složitosti

Platí-li definice 2.5.4, říkáme:

časové složitosti, které se liší pouze multiplikativní konstantou
patří do *stejně třídy* časové složitosti.

Blok 2.5.5: Třída časové složitosti

Přesněji bychom měli zapisovat:

$$f(n) \in \mathcal{O}(g)$$

a číst:

$$f \text{ je třídy } \mathcal{O}(g)$$

Poznámka:

Symbol „ $\in$ “ v definici 2.5.4 by měl být spíše nahrazen symbolem „ $\preceq$ “ ve významu:

je asymptoticky rovno.

To, že funkce $c \cdot g(n)$ pro $n > n_0$ nikdy není menší než funkce $f(n)$, viz definice 2.5.4, také znamená, že odhad časové složitosti je pro daný algoritmus ten *nejhorší možný*. Jinými slovy to znamená, že praktický výpočet pro zvolený algoritmus *nikdy* nedopadne hůře, spíše lépe.

Obecné stanovení časové složitosti algoritmu je poměrně obtížná úloha. Různých druhů algoritmů je přinejmenším tolik, kolik je různých druhů úloh (technických, matematických, fyzikálních, biologických, sociologických, ...). Proto existuje značné množství hodnotících postupů, každý má nějaké „přednosti“ pro určité typy úloh, pro jiné nikoliv.

V prostoru vymezeném tímto sešitem bohužel není možné se věnovat všem aspektům a metodám, které lze použít pro stanovení časové složitosti. Vnímavý čtenář bude jistě schopen dohledat potřebné odborné pasáže.

Kapitola 3

Zákony přírody - symboly poznání

Obsah kapitoly

3.1	Rovnice a rovnosti	36
3.1.1	Výlet do školních lavic	36
3.1.2	Lineární (algebraická) rovnice pro jednu neznámou	37
3.1.3	Polynomiální rovnice pro jednu neznámou	38
3.1.4	Obecná rovnice pro jednu neznámou	39
3.2	Jedna rovnice pro více neznámých	41
3.2.1	Jedna rovnice pro dvě neznámé	41
3.2.2	Funkcionální rovnice - globální a lokální řešení	43
3.3	Rovnice z geometrie	45
3.3.1	Rovnice křivek a ploch	45
3.3.2	Soustava dvou rovnic pro dvě neznámé. Grafický a ...	47
3.4	Rovnice lineární algebry	48
3.4.1	Soustava dvou rovnic pro tři neznámé	48
3.4.2	Základní úloha lineární algebry	50

Motto: - Máš-li problém, formuluj jej matematicky;
- matematický problém převed' na algebraickou úlohu;
- algebraickou úlohu přeformuluj na algebraickou rovnici;
- rovnici vyřeš.

Máš po problému.

Název tohoto odstavce jsme opsali z knihy Sander Bais: De natuurwetten. Iconen van onze kennis. Amsterdam 2005. Existuje český překlad z roku 2009 s názvem „Rovnice - symboly poznání“.

Vzpomeneme-li si ještě na knihu R.E.Peierls: Zákony přírody (Orbis Praha 1961), máme zde dva prameny vzdělání, které by neměly chybět v knihovně učitele.

3.1 Rovnice a rovnosti

Rovnost je vztah (relace)! Rovnice je úloha! Podobně jako nerovnost a nerovnice. *Matematickým modelem* nějakého zákona je rovnost nebo nerovnost (případně jejich soubor). Je jistě pozoruhodné, že v technologiích se zákonům (vztahům) velmi často říká rovnice a fakticky jde o rovnosti.

Například: pohybová rovnice, rovnice kontinuity, stavová rovnice, Maxwellovy rovnice, rovnice vedení tepla, Navierovy-Stokesovy rovnice, Schrodingerova rovnice, atd. Fakticky máme pohybový zákon, zákon spojitosti, stavový zákon, Maxwellovy zákony, zákon vedení tepla, Navierův-Stokesův zákon, Schrodingerův zákon, atd.

Jsou ovšem čestné výjimky: Archimedův zákon, Ampérův zákon, zákon nabídky a poptávky, atd. Ty jsou ale dané spíše z historických důvodů. Je pozoruhodné, že se neříká Coulombova rovnice, Ohmova rovnice, Kirchhoffova rovnice, ...

Již ve starověku se formulovaly úlohy, které měly čistě matematický charakter. Jediným nástrojem k jejich řešení byl mozek (úsudkové úlohy, slovní úlohy). Tendence lidí vylepšovat svoje nástroje vedly k rozvoji abstraktních (matematických) nástrojů a technologií.

3.1.1 Výlet do školních lavic

Při vzpomínkách na hodiny matematiky se většině z nás asi vybaví všelijaké nudné manipulace s písmeny a čísly (na papíře nebo na tabuli) v rámci nějakých úkolů či příkladů. Například:

$$\text{Řešte rovnici: } (12x + 3)^2 + (5x + 3)^2 = (13x + 4)^2 !$$

Žák: Co se po mně chce? Co mám dělat?

Pedagog: Musíš najít x !

Žák: Proč ho mám hledat? Vždyť je to tady napsaný a dokonce třikrát.

Pedagog: Musíš to x vypočítat, abys znal řešení rovnice!

Žák: (Já to znát nepotřebuji, vy to chcete!) A co mám tedy dělat?

Pedagog: Nejdříve musíš upravit výrazy v závorkách podle vzorečku:

$$(a + b)^2 = a^2 + 2ab + b^2 !$$

Žák: Je to takto dobře? $144x^2 + 72x + 9 + 25x^2 + 30x + 9 = 169x^2 + 104x + 16$;

Pedagog: Ano. Teď na každé straně rovnice sečti, co se dá sečíst

Žák: $169x^2 + 102x + 18 = 169x^2 + 104x + 16$; a co dál?

Pedagog: Teď musíš všechny členy na pravé straně převést na levou stranu a sečíst, co jde sečíst!

Žák: $-2x + 2 = 0$; je to dobře?

Pedagog: Výborně! Teď si vzpomeň na vzoreček, který jsem vám minule nadiktoval do sešitu:

$$ax + b = 0, x = -\frac{b}{a}.$$

Žák: $x = -\frac{(-2)}{2} = 1$, je to dobře?

Pedagog: Výborně. Vyřešil jsi danou rovnici.

Žák: (No a co? K čemu to budu potřebovat?)

Pedagog: (tuší, co se žákovi honí hlavou) Podívej se, chceš-li hrát profesionálně fotbal, musíš hodně trénovat – běhat, skákat, skákat na každé noze zvlášť, házet, trénovat obratnost a manipulaci s míčem, atd. To všechno budeš ve fotbale potřebovat.

Žák: Já ale matiku nikdy potřebovat nebudu!

Končíme známým povzdechem pedagoga:

- Když ořu na poli, vidím za sebou brázdu, když ořu s vámi, nevidím za sebou nic!

→ 3.1.2 Lineární (algebraická) rovnice pro jednu neznámou

Také můžeme říkat: Algebraická rovnice 1.stupně (neznámá se vyskytuje v 1.mocnině). Předpokládáme, že čtenář je obeznámen s tímto tématem na úrovni příruček typu: J.Polák: Středoškolská matematika v úlohách.

Přesto začneme výklad nejjednodušší rovnicí, abychom „se odrazili“ k výkladu složitějších rovnic.

Formulace:

Jsou dána čísla $a \in \mathbb{C}$, $b \in \mathbb{C}$. Chceme najít číslo $\bar{x} \in \mathbb{C}$ takové, že platí rovnost:

$$a\bar{x} + b = 0.$$

Číslo $\bar{x}$ říkáme *řešení (kořen) lineární rovnice*: $ax + b = 0$.

Pokyn „Řešte v $\mathbb{C}$ rovnici $ax + b = 0$ “ je přijatelná formulace úlohy, pokud si uvědomujeme, že jde o zkratku. Pokyn „Řešte v $\mathbb{R}$ rovnici $ax + b = 0$ “ znamená, že řešení, kořen(y) hledáme pouze v $\mathbb{R}$.

Řešitelnost:

1. Když $a \neq 0$, potom číslo $\bar{x} = -b/a$ je jediným řešením.
2. Když $a = 0$, $b = 0$, potom řešením je libovolné číslo z $\mathbb{C}$. Říkáme také, že rovnice má nekonečně (nespočetně) mnoho řešení.
Řešíme-li rovnici na podmnožině $\mathbb{C}$, pak toto tvrzení nemusí platit. Viz diofantické rovnice, odst. 1.2.2.
3. Když $a = 0$, $b \neq 0$, potom rovnice je v $\mathbb{C}$ neřešitelná (nemá v $\mathbb{C}$ řešení).

Blok 3.1.1: Lineární rovnice pro jednu neznámou

Metody - postupy vedoucí k dosažení řešení:

- a) Algebraické manipulace a operace: viz výše zmíněné příručky.
- b) Geometrické (grafické) metody: viz tamtéž.
- c) Iterační metody: uvedeme princip, který budeme později potřebovat.

Rovnost:

$$a\hat{x} + b = 0$$

upravíme na:

$$\hat{x} + a\hat{x} = \hat{x} - b,$$

pak například bud:

$$\hat{x} = (1 - a)\hat{x} - b,$$

nebo:

$$\hat{x} = \frac{\hat{x} - b}{1 + a};$$

(je nekonečně mnoho takových možností).

Zvolíme libovolné číslo x_0 a počítáme posloupnost podle rekurentní formule:

$$x_{n+1} = (1 - a)x_n - b,$$

nebo formule:

$$x_{n+1} = \frac{x_n - b}{1 + a}.$$

Když posloupnost konverguje, tak konverguje ke kořenu $\hat{x}$.

Poznámka - především didaktická:

Nejen matematiky vždy inspirovala otázka neřešitelnosti daného problému (úlohy). To vedlo k rozšiřování číselných množin. Vyjasňování možností „dělení nulou“ vedlo také k *limitám* a v abstraktní algebře k tzv. *dělitelům nuly*.

3.1.3 Polynomiální rovnice pro jednu neznámou

Formulace:

Je dán polynom (mnohočlen) n – tého stupně v proměnné x :

$$P_n(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \dots + a_1 \cdot x + a_0, \quad a_n \neq 0, \quad n \in \mathbb{N}, \quad a_i \in \mathbb{C}, \quad i = 0, 1, 2, \dots, n.$$

Chceme najít číslo $\hat{x} \in \mathbb{C}$ takové, že platí rovnost:

$$P_n(\hat{x}) = 0.$$

Číslu $\hat{x}$ říkáme *řešení (kořen)* rovnice: $P_n(x) = 0$.

Řešitelnost: Připomeňme pouze, že v $\mathbb{C}$ existuje právě n kořenů, a že komplexní kořeny tvoří komplexně sdružené dvojice právě když daný polynom má všechny koeficienty reálné!

Řešením je tedy n -tice kořenů.



Blok 3.1.2: Polynomiální rovnice

Metody: Celé generace matematiků se pokoušely najít postupy, jak pomocí manipulací s odmocninami najít (vypočítat) všechny kořeny.

Výsledky jsou sice poměrně chabé (s výjimkou tzv. Cardanových vzorců), ale je to bohatě vyváženo rozvojem abstraktní algebry s odpověďmi na mnohem obecnější otázky.

Pro kvadratickou rovnici máme metodu doplnění na čtverec která nám odmocňováním poskytne známé vzorce.

V technologiích jsou (zatím) plně vyhovující numerické metody.

3.1.4 Obecná rovnice pro jednu neznámou

Formulace úlohy v $\mathbb{R}$:

Je dán algebraický (funkcionální) výraz $f(x)$ v proměnné x . Výraz $f(x)$ chápeme jako funkci:

$$f : \mathcal{D} \rightarrow \mathcal{H},$$

kde $\mathcal{D} \subset \mathbb{R}$ je definiční obor dané funkce a $\mathcal{H} \subset \mathbb{R}$ je její obor hodnot. Dále je dáno číslo $\hat{b} \in \mathbb{R}$ a množina $\mathcal{W} \subset \mathbb{R}$ ($\mathcal{W}$ může představovat speciální množinu, v níž hledáme kořen; například všechna kladná čísla).

Chceme najít číslo $\hat{x} \in \mathcal{W}$ takové, aby platila rovnost:

$$f(\hat{x}) = b.$$

Číslo $\hat{x}$ říkáme *kořen* nelineární rovnice $f(x) = b$.

Množině všech kořenů se také říká *řešení* (angl. solution) rovnice.

Poznamenejme, že formule:

$$f(x) = b$$

je pouze stručným zápisem formulované úlohy.

V praxi se častěji setkáváme se zápisem:

$$f_1(x) - f_2(x) = b$$

kde f_1 a f_2 jsou dané funkce.

Řešitelnost úlohy:

1. Když $b \in \mathcal{H}$ (b patří do oboru hodnot funkce f), potom úloha má aspoň jeden kořen.
2. Když $b \in \mathcal{H}$ a funkce f je ostře monotónní, potom úloha má právě jeden kořen v $\mathcal{D}$.
3. Když $b \notin \mathcal{H}$, potom úloha je neřešitelná, nebo nemá smysl. Sem patří slavné úlohy:

$$x^2 = -1, \quad \sin x = 5 \quad (\text{úloha dělostřelců za Velké vlastenecké války})$$

Blok 3.1.3: Rovnice v $\mathbb{R}$ 

Metody: Algebraickými manipulacemi s výrazem $f(x)$ lze uspět pouze u specifických typů rovnic. Obvykle používáme převod na jednodušší rovnice. Musíme však pečlivě analyzovat, jestli upravená rovnice má stejná řešení (kořeny) jako rovnice původní.

Metody pro středoškolské verze těchto rovnic (goniometrické, exponenciální, logaritmické, iracionální) se opírají o převody na jednoduchý typ rovnic, u nichž se výsledek „snadno uhádne“. Například rovnice:

$$\sin x + \cos x = 1$$

se převede na rovnici:

$$\sin\left(x + \frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}.$$

Výsledek formálně zapíšeme:

$$\hat{x} = \arcsin\left(\frac{\sqrt{2}}{2}\right) - \frac{\pi}{4}.$$

Skoro univerzálními metodami jsou metody numerické. Zde však těmto metodám nechceme věnovat větší pozornost. Uvedme aspoň jejich názvy: metoda prosté iterace, bisekce, regula falsi, tečen, sečen, Newtonova, Mullerova, Graeffova, Bairstowova.

Poznámka: Matematický profesionál nebo aspoň poloprofesionál by si měl uvědomit („amatéři“ si toho většinou vůbec nevšimnou) jednu jazykově-filozoficko-výkladovou potíž související s užíváním symbolu „=“. Rovnice:

$$f(x) = 0$$

je symbolický zápis úlohy.

Například máme tuto úlohu: Najdi číslo, které po vynásobení tímtož číslem, přičtením pětinasobku hledaného čísla a nakonec odečtením 14, dává nulu.

Jak teď vysvětlíme tzv. úpravy úlohy (rovnice). Vždyť algebraické úpravy se týkají rovností (!). Úloha může být neřešitelná, ale my jí „vesele upravujeme“.

Úprava, která z neřešitelné úlohy dělá úlohu řešitelnou, je ekvivalentní? A co úprava, která z neřešitelné úlohy dělá opět úlohu neřešitelnou? Ano, důvěřuj ale prověřuj!

U celé řady složitějších úloh se tzv. zkouška dosazením nedá udělat nebo vůbec nemá smysl.

Proto formulace rovnic v našem výkladu vypadají poněkud komplikovaně. Nechceme „nosit dříví do lesa“, ale přimlouváme se, aby formulace:

řešme rovnici...

se přeložila takto:

Předpokládejme, že existuje řešení $\hat{x}$ a platí rovnost $f(\hat{x}) = 0$. Tuto rovnost budeme algebraicky upravovat..., atd.

3.2 Jedna rovnice pro více neznámých

3.2.1 Jedna rovnice pro dvě neznámé

Mějme (lineární) rovnici $F(x, y) = 3x + 4y - 1 = 0$.

Elementární poznatky:

- Víme, že každá dvojice čísel typu $(x, y) : (\frac{1}{3}, 0), (0, \frac{1}{4}), (-1, 1)$, atd. splňuje rovnost $3x + 4y - 1 = 0$.
- Takové dvojici říkáme *řešení* rovnice $3x + 4y - 1 = 0$.
- Víme také, že „technologickými manipulacemi“ můžeme „vypočítat“:

$$y = \frac{(1 - 3x)}{4}, \quad \text{nebo} \quad x = \frac{(1 - 4y)}{3}.$$

Pak také můžeme kterékoliv řešení dané rovnice zapsat ve tvaru dvojice:

$$(x, \frac{(1 - 3x)}{4}) \quad \text{nebo dvojice} \quad (\frac{(1 - 4y)}{3}, y)$$

a konkrétní dvojici vypočteme volbou x v prvním případě nebo volbou y v druhém případě. Tyto dvojice pak určují vztah (relaci, přiřazení, funkci) mezi hledanými (neznámými) parametry x, y . Tento vztah je tedy danou rovnicí určen skrytě (implicitně).

- Víme také, že volbu buď x , nebo y můžeme realizovat „složitěji“, např.:

$$x = -1 + t;$$

pak dopočteme:

$$y = \frac{1 - 3t}{4}$$

a vidíme, že dvojice $(-1 + t, \frac{1 - 3t}{4})$ reprezentuje všechna řešení dané rovnice pro libovolnou hodnotu parametru t .

- Víme také, že když v kartézském (pravoúhlém) souřadnicovém systému přiřadíme každému řešení (dvojici) *bod v rovině*, množina těchto bodů vytvoří přímku a daná rovnice se pak nazývá rovnice přímky a vztahy:

$$x = -1 + t, \quad y = \frac{1 - 3t}{4},$$

se nazývají *parametrické rovnice přímky*.

Obecný zápis lineární rovnice pro dvě neznámé: (a, b, c jsou daná čísla)

$$ax + by + c = 0.$$

Analogicky:

Obecný zápis lineární rovnice pro n neznámých: ($a_i, i = 1, 2, \dots, n, c$ jsou daná čísla)

$$a_1x_1 + a_2x_2 + \dots + a_nx_n + c = 0.$$

Řešením je n -tice (množina n -tic) čísel $(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$, které která splňují rovnost $a_1\bar{x}_1 + a_2\bar{x}_2 + \dots + a_n\bar{x}_n + c = 0$.

Blok 3.2.4: Lineární rovnice pro více neznámých

Úloha:

Mějme rovnici $y^2 - x = 0$. Zde $F(x, y) = y^2 - x$.

Řešení:

Vidíme, že řešením této rovnice jsou funkce:

$$y = \sqrt{x}, \quad y = -\sqrt{x}; \quad x \geq 0$$

tj. množiny dvojic: $(x, \sqrt{x})$, $(x, -\sqrt{x})$, (části paraboly), viz obrázek 3.2.4.

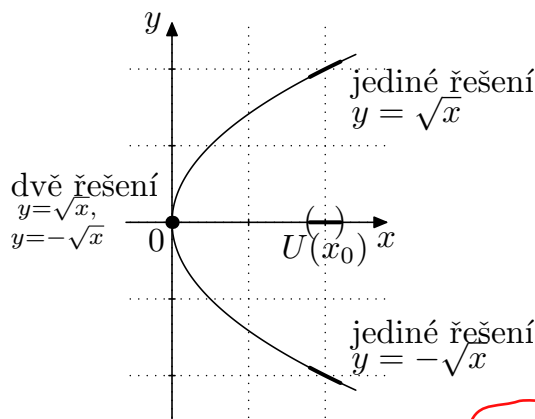
Pro $x < 0$ není rovnice řešitelná v $\mathbb{R}$, tj. neexistuje reálná funkce $y = f(x)$, která by splňovala rovnost $F(x, y) = 0$.

Pro $x \geq 0$ je rovnice řešitelná (dvě řešení - funkce).

Pro $x \geq 0, y \geq 0$ je rovnice řešitelná jednoznačně, tj. mezi nezápornými funkcemi existuje jediné řešení.

V okolí $U(x_0)$ bodu x_0 je funkce $y = +\sqrt{x}$ jediným lokálním řešením rovnice a funkce $y = -\sqrt{x}$ je také jediným lokálním řešením dané rovnice. Kdežto v okolí bodu 0 neexistuje funkce $y = f(x)$, která by splňovala danou rovnici.

Rozpaky čtenáře nad těmito tvrzeními vyplývají z toho, že si neuvědomuje definici funkce, tj., že každému $x \in U(x_0)$ musí být přiřazeno jediné číslo y .



Obrázek 3.2.1: Řešení funkcionální úlohy $x - y^2 = 0$

3.2.2 Funkcionální rovnice - globální a lokální řešení

Formulace úlohy v $\mathbb{R}^2$ - rovnice pro dvě neznámé:

Je dán funkcionální výraz $F(x, y)$ a množina $\mathcal{D} \subset \mathbb{R}^2$.

Výraz $F(x, y)$ chápeme jako reálnou funkci: $F: \mathcal{D} \rightarrow \mathcal{H}$, kde: $\mathcal{D}$ je definiční obor a $\mathcal{H} \subset \mathbb{R}$ je obor hodnot.

Chceme najít dvojici čísel $(\hat{x}, \hat{y}) \in \mathcal{D}$ takovou, pro které platí rovnost $F(\hat{x}, \hat{y}) = 0$.

Dvojice čísel $(\hat{x}, \hat{y}) \in \mathcal{D}$ se nazývá **řešení rovnice** $F(x, y) = 0$ (kořen rovnice) a určuje bod rovině $\mathbb{R}^2$.

Množina **všech** takových bodů (dvojic čísel) $\mathcal{M} = \{(\hat{x}, \hat{y}) \in \mathcal{D} : F(\hat{x}, \hat{y}) = 0\}$ se také nazývá **řešení rovnice**.

Zápis úlohy:

$$F(x, y) = 0.$$

Blok 3.2.5: Funkcionální rovnice v $\mathbb{R}^2$

Definujeme:

- (A) Funkce $y = f(x)$, $x \in \mathcal{J} \subset \mathbb{R}$ je na $\mathcal{J}$ *globálním řešením* rovnice $F(x, y) = 0$, jestliže:

$$\forall x \in \mathcal{J} \text{ platí rovnost } F(x, f(x)) = 0,$$

Platí-li požadovaná rovnost pouze v okolí nějakého bodu x_0 , tj.:

$$\forall x \in \mathcal{U}(x_0) \subset \mathcal{J} : F(x, f(x)) = 0,$$

říkáme, že funkce $y = f(x)$ je *lokálním řešením* (v okolí x_0) dané rovnice.

- (B) Funkce $z = f(x, y)$, $(x, y) \in \Omega \subset \mathbb{R}^2$ je na Ω *globálním řešením* rovnice o třech neznámých $F(x, y, z) = 0$ jestliže:

$$\forall (x, y) \in \Omega \text{ platí rovnost } F(x, y, f(x, y)) = 0.$$

Platí-li požadovaná rovnost pouze v okolí bodu (x_0, y_0) , tj.:

$$\forall (x, y) \in \mathcal{U}(x_0, y_0) \subset \Omega,$$

říkáme, že funkce $z = f(x, y)$ je *lokálním řešením* dané rovnice.

- (C) Funkce $u = f(\mathbf{x})$, $\mathbf{x} \in \Omega \subset \mathbb{R}^n$ je na Ω *globálním řešením* rovnice o $n+1$ neznámých $F(\mathbf{x}, u) = 0$, jestliže:

$$\forall \mathbf{x} \in \Omega \text{ platí rovnost } F(\mathbf{x}, f(\mathbf{x})) = 0; \mathbf{x} = [x_1, x_2, \dots, x_n]^T.$$

Platí-li požadovaná rovnost pouze v okolí bodu $\mathbf{x}^{(0)}$, tj.:

$$\forall \mathbf{x} \in \mathcal{U}(\mathbf{x}^{(0)}) \subset \Omega,$$

říkáme, že funkce $u = f(\mathbf{x})$ je *lokálním řešením* dané rovnice.

Blok 3.2.6: Globální a lokální řešení

Teoretická analýza řešitelnosti funkcionálních rovnic je velmi rozsáhlá a vyžaduje velmi hluboké poznatky z matematické a funkcionální analýzy, diferenciální geometrie a z teorie křivek a ploch. Připomeňme pouze, že řešitelnost integrálních a diferenciálních rovnic se o tuto analýzu opírá.

Na tomto místě uvedeme jen několik praktických poznatků na ilustrativních příkladech. Další poznatky geometrického typu jsou uvedeny v odst. 3.3.

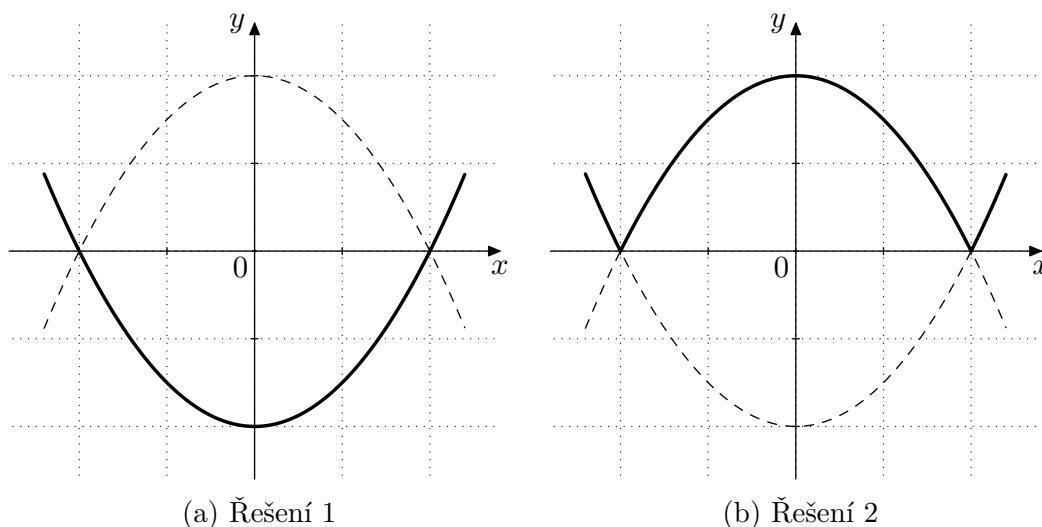
Například rovnice

$$y^2 - (x^2 - 1)^2 = 0$$

má v $\mathbb{R}$ osm spojitých řešení $y = f(x)$:

- | | |
|---|---|
| 1. $y = x^2 - 1$, | 6. $y = \begin{cases} 1 - x^2, & \text{pro } (-\infty, -1), \\ x^2 - 1, & \text{pro } (-1, +\infty), \end{cases}$ |
| 2. $y = 1 - x^2$, | |
| 3. $y = x^2 - 1 $, | 7. $y = \begin{cases} x^2 - 1, & \text{pro } (-\infty, 1), \\ 1 - x^2, & \text{pro } (1, +\infty), \end{cases}$ |
| 4. $y = - 1 - x^2 $, | |
| 5. $y = \begin{cases} x^2 - 1, & \text{pro } (-\infty, -1), \\ 1 - x^2, & \text{pro } (-1, -\infty), \end{cases}$ | 8. $y = \begin{cases} 1 - x^2, & \text{pro } (-\infty, 1), \\ x^2 - 1, & \text{pro } (1, +\infty). \end{cases}$ |

Viz obrázky 3.2.3a,b.



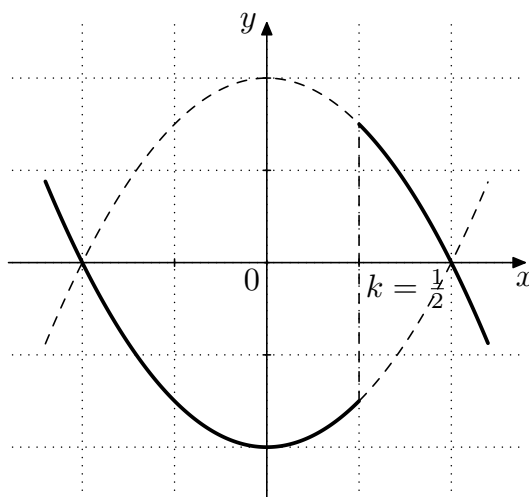
Obrázek 3.2.3: Dvě ze spojitých řešení rovnice $y^2 - (x^2 - 1)^2 = 0$

A nekonečně mnoho nespojitých řešení typu :

$$y_k(x) = \begin{cases} x^2 - 1, & \text{pro } \forall x < k, \\ 1 - x^2, & \text{pro } \forall x \geq k, \end{cases} \quad k \geq 0,5$$

Viz obrázek 3.2.4.

Poznámka: Spolehnáme na to, že čtenář aspoň tuší, co je spojitá, resp. nespojitá funkce.



Obrázek 3.2.4: Jedno nespojitě řešení rovnice $y^2 - (x^2 - 1)^2 = 0$; $k = \frac{1}{2}$

3.3 Rovnice z geometrie

3.3.1 Rovnice křivek a ploch

Připomínáme známá (snad) fakta ze střední školy. Mějme však na paměti (!), že proměnné (neznámé) interpretujeme v pravoúhlé soustavě souřadnic.

Rovnice přímky

$$ax + by + c = 0, \quad (a, b) \neq (0, 0),$$

čísla a, b, c jsou dána, x, y jsou proměnné (neznámé).

Řešení rovnice přímky, tj. množina

$$p = \{(x, y) \in \mathbb{R}^2 : ax + by + c = 0\},$$

se nazývá **přímka** v $\mathbb{R}^2$. Tedy přímka je řešením rovnice přímky.

Blok 3.3.7: Přímka

Rovnice kružnice

$$(x - x_0)^2 + (y - y_0)^2 = r^2,$$

x_0, y_0, r jsou daná čísla, x, y jsou proměnné (neznámé).

Řešení rovnice kružnice, tj. množina

$$\mathcal{K} = \{(x, y) \in \mathbb{R}^2 : (x - x_0)^2 + (y - y_0)^2 = r^2\},$$

se nazývá **kružnice** v $\mathbb{R}^2$ se středem $S = (x_0, y_0)$ a poloměrem $r > 0$.

Tedy kružnice je řešením rovnice kružnice.

Blok 3.3.8: Kružnice

Rovnice elipsy

$$\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1,$$

$x_0, y_0, a > 0, b > 0$ jsou daná čísla, x, y jsou proměnné (neznámé).

Řešení rovnice elipsy, tj. množina

$$\mathcal{E} = \{(x, y) \in \mathbb{R}^2 : \frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1\},$$

se nazývá **elipsa** v $\mathbb{R}^2$ se středem $S = (x_0, y_0)$ a poloosami $a > 0, b > 0$.

Blok 3.3.9: Elipsa

Rovnice roviny

$$ax + by + cz + d = 0, \quad (a, b, c) \neq (0, 0, 0),$$

čísla a, b, c jsou dána, x, y, z jsou proměnné (neznámé).

Řešení rovnice roviny, tj. množina

$$P = \{(x, y, z) \in \mathbb{R}^3 : ax + bx + cz + d = 0\},$$

se nazývá **rovina** v $\mathbb{R}^3$. Tedy rovina je řešením rovnice roviny.

Blok 3.3.10: Rovina

Znázorněte v rovině $\mathbb{R}^2$ množiny $p, \mathcal{K}, \mathcal{E}$ z předchozího výkladu, případně **další (!)** množiny jako řešení příslušných rovnic:

$\mathcal{D}_1 = \{(x, y) \in \mathbb{R}^2 : y - x^2 = 0\}$	- (parabola)
$\mathcal{D}_2 = \{(x, y) \in \mathbb{R}^2 : x^2 - 2y^2 = 0\}$	- (hyperbola)
$\mathcal{D}_3 = \{(x, y) \in \mathbb{R}^2 : xy = 1\}$	- (hyperbola)
$\mathcal{D}_4 = \{(x, y) \in \mathbb{R}^2 : y - e^x = 0\}$	- (exponenciála)
$\mathcal{D}_5 = \{(x, y) \in \mathbb{R}^2 : y - \sin x = 0\}$	- (sinusoida)
$\mathcal{D}_6 = \{(x, y) \in \mathbb{R}^2 : y - \sqrt{1 - x^2} = 0\}$	- (půlkružnice)
$\mathcal{D}_7 = \{(x, y) \in \mathbb{R}^2 : y^2 + x^2 = 1\}$	- (kružnice)

Příklad 3.3.1: příklady jednoduchých křivek

Poznámka: Geometrické (grafické) znázornění množiny řešení $\mathcal{D}$, se také nazývá **graf rovnice**.

Závěr: Nemáme ambice v tomto odstavci vyložit nějakou podstatnou část analytické geometrie křivek a ploch, či dokonce část diferenciální geometrie. Chtěli jsme pouze, v duchu našeho výkladu o rovnicích, upozornit na některé souvislosti a pojmy. Například:

1. Funkce $y = f(x)$ může být řešením algebraické rovnice typu $F(x, y) = 0$; pak se tato funkce často nazývá *implicitní funkcí* nebo funkcí určenou rovnicí implicitně.
2. Tzv. *parametrické rovnice* křivky jsou z našeho pohledu zápisem (globálního) řešení příslušné rovnice křivky.

3.3.2 Soustava dvou rovnic pro dvě neznámé. Grafický a numerický přístup

Mějme soustavu dvou funkcionálních rovnic pro dvě neznámé:

$$\begin{aligned}x^2 + y^2 - x &= 0, \\x^2 - y^2 - y &= 0.\end{aligned}$$

První je rovnice kružnice a jejím globálním řešením je kružnice. Druhá je rovnicí hyperboly a jejím globálním řešením je hyperbola (viz obrázek 3.3.5). Algebraicky jsou to dvě kvadratické rovnice pro dvě neznámé.

Lokální řešení první rovnice jsou dvě:

$$\begin{aligned}y &= \sqrt{\frac{1}{4} - \left(x - \frac{1}{4}\right)^2}; \quad x \in \langle 0, 1 \rangle \quad \text{„horní půlkružnice“} \\y &= -\sqrt{\frac{1}{4} - \left(x - \frac{1}{4}\right)^2}; \quad x \in \langle 0, 1 \rangle \quad \text{„dolní půlkružnice“}\end{aligned}$$

Lokální řešení druhé rovnice jsou také dvě:

$$\begin{aligned}y &= \frac{-1 + \sqrt{1 + 4x^2}}{2}; \quad x \in \mathbb{R} \quad \text{horní větev paraboly} \\y &= \frac{-1 - \sqrt{1 + 4x^2}}{2}; \quad x \in \mathbb{R} \quad \text{dolní větev paraboly}\end{aligned}$$

Chceme najít takovou dvojici bodů $(\hat{x}, \hat{y}) \in \Omega \subset \mathbb{R}^2$, aby příslušné rovnosti platily současně, tj. aby:

$$\hat{x}^2 + \hat{y}^2 - \hat{x} = 0 \quad \wedge \quad \hat{x}^2 - \hat{y}^2 - \hat{y} = 0;$$

Geometricky hledáme průsečíky uvedených křivek.

Algebraická metoda: Na eliminačním principu dostaneme polynomiální rovnici 4.stupně.

Grafická metoda: viz obrázek 3.3.5. Zjistíme, že jedna dvojice čísel je (0,0) a druhá (0,8;0,4) - odhad.

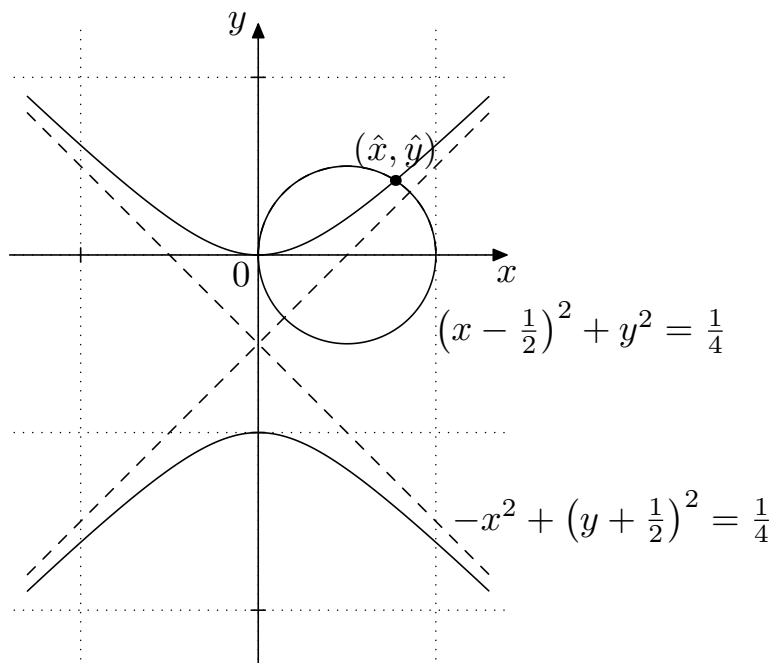
Numerická metoda: Počáteční aproximaci (0,8;0,4) zpřesníme Newtonovou iterační metodou s přesností 10^{-7}):

$$\hat{x} \doteq 0,7718445; \quad \hat{y} \doteq 0,4196434.$$

Dvojice $(\hat{x}, \hat{y})$, se nazývají **řešení soustavy rovnic**

$$\begin{aligned}\hat{x}^2 + \hat{y}^2 - \hat{x} &= 0 \\ \hat{x}^2 - \hat{y}^2 - \hat{y} &= 0\end{aligned}$$

Takových dvojic lze často nalézt více, než jednu. Množina **všech** takových dvojic se také nazývá **řešení soustavy rovnic**.



Obrázek 3.3.5: Grafické řešení soustavy rovnic

3.4 Rovnice lineární algebry

Předem se omlouváme, ale bez přípravy skočíme do technologie maticového počtu. Trocha přípravy je uvedena v prvním sešitu, odstavci 4.3. Zobrazení množin.

Základním technologickým principem je eliminace neznámých. Říkáme, že „upravujeme soustavu rovnic“, ale fakticky pracujeme s danými koeficienty soustavy, tj. s maticí **A** a sloupcem **b**.

3.4.1 Soustava dvou rovnic pro tři neznámé

Mějme soustavu rovnic lineárních:

$$\begin{aligned}x_1 - 4x_2 + 7x_3 &= 10 \\ x_2 - 2x_3 &= -4.\end{aligned}$$

$$\mathbf{Ax} = \mathbf{b} \Leftrightarrow \begin{bmatrix} 1 & 4 & 7 \\ 0 & 2 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 10 \\ -4 \end{bmatrix}$$

Nejdříve se na tuto úlohu podívejme pohledem odstavce 3.2.2. Globálním řešením je množina všech trojic (x_1, x_2, x_3) splňujících dané požadavky. Z první rovnice vidíme, že x_1 závisí na x_2, x_3 , tj. existuje funkce f_1 taková, že:

$$x_1 = f_1(x_2, x_3) = 10 + 4x_2 - 7x_3;$$

z druhé rovnice vidíme, že:

$$x_2 = f_2(x_3) = -4 + 2x_3.$$

Takže: $x_1 = f_1(f_2(x_3), x_3)$, a potom řešení (trojice) může být zapsáno :

$$(x_1, x_2, x_3) = (f_1(f_2(x_3), x_3); f_2(x_3); x_3),$$

tedy jako funkce jedné proměnné (parametru) x_3 . V tomto případě jde o funkci o třech složkách, tedy o vektorovou funkci.

Přeložíme popsané úvahy do srozumitelnější podoby:

z druhé rovnice jsme stanovili x_2 v závislosti na x_3 ; z první rovnice x_1 v závislosti na x_2 a x_3 . Vlastně jsme realizovali eliminační proceduru.

Nyní konkrétní postup:

a) ruční:

$$x_3 = t$$

a z 2.rovnice vypočteme:

$$x_2 = -4 + 2t$$

a z 1.rovnice vypočteme:

$$x_1 = -6 + t.$$

Zapíšeme výsledek ve sloupcích:

$$\mathbf{x} = \mathbf{v} + t \cdot \mathbf{u}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} -6 \\ -4 \\ 0 \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

kde platí:

$$\mathbf{Ax} = \mathbf{Av} = \mathbf{b}, \quad \mathbf{Au} = \mathbf{0}.$$

- $\mathbf{x}$ nazýváme obecné řešení soustavy,
- $\mathbf{v}$ nazýváme partikulární řešení soustavy,
- $t \cdot \mathbf{u}$ nazýváme obecné řešení příslušné homogenní soustavy.

Geometricky je obecné řešení parametrickým vyjádřením průsečnice dvou rovin určených danou soustavou (tzv. vektorové vyjádření přímky).

b) algoritmický, určený pro „velké soustavy“. Tu si představíme jako maticově zapsanou ve tvaru:

$$\begin{bmatrix} 1 & 4 & 7 \\ 0 & 1 & -2 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 10 \\ -4 \\ 0 \end{bmatrix}$$

Abychom mohli stanovit všechna řešení dané soustavy, sestavíme si maticovou rovnici:

$$\begin{bmatrix} 1 & 4 & 7 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 & u_1 \\ v_2 & u_2 \\ v_3 & u_3 \end{bmatrix} = \begin{bmatrix} 10 & 0 \\ -4 & 0 \\ 0 & 1 \end{bmatrix}$$

Nulu na diagonále matice soustavy odpovídající neznámé x_3 nahradíme volbou $t = 1$ a k původní pravé straně připojíme vektor $[0, 0, 1]^T$. Řešením získané maticové rovnice je matice:

$$\mathbf{X} = \begin{bmatrix} -6 & 1 \\ -4 & 2 \\ 0 & 1 \end{bmatrix}$$

jejíž sloupce určují všechna řešení původní (nedourčené) soustavy, tj.

$$\mathbf{x} = [-6, -4, 0]^T + t[1, 2, 1]^T = \mathbf{u} + t \cdot \mathbf{v}.$$

Laskavý čtenář se může dosazením přesvědčit, že $\mathbf{v}$ je řešením původní nehomogenní soustavy a $\mathbf{u}$ řešením příslušné homogenní soustavy.

Vektor $\mathbf{x} = \mathbf{v} + t \cdot \mathbf{u}$ je tzv. *obecné řešení* soustavy. Předností tohoto postupu je, že můžeme užít standardní algoritmy pro regulární matici soustavy.

Blok 3.4.11: Ilustrace metody řešení „nedourčené“ soustavy

3.4.2 Základní úloha lineární algebry

Zde potřebujeme, aby čtenář měl alespoň základní představu o Eukleidovském prostoru $\mathbb{E}^n$ se skalárním součinem (viz např. Sešit I, odst. 6.2.2.).

Základní úloha lineární algebry:

Je dána matice $\mathbf{A}$ a sloupec $\mathbf{b}$.

Stanovte sloupec $\hat{\mathbf{x}}$ takový, aby platila rovnost: $\mathbf{A}\hat{\mathbf{x}} = \mathbf{b}$.

Tuto úlohu zapisujeme stručně $\mathbf{A}\mathbf{x} = \mathbf{b}$ a nazýváme *soustava lineárních algebraických rovnic*. Sloupec $\hat{\mathbf{x}}$ se nazývá řešení úlohy.

Existuje-li řešení $\hat{\mathbf{x}}$ této úlohy, potom pro matici $\mathbf{A}$ typu (m, n) a sloupec $\hat{\mathbf{x}}$ typu $(n, 1)$ musí být sloupec $\mathbf{b}$ typu $(m, 1)$ (nutná podmínka řešitelnosti).

Jsou to nutné podmínky, nikoliv však postačující podmínky řešitelnosti. Neznamena to, že když $\mathbf{A}, \mathbf{b}$ jsou správného typu, že je úloha řešitelná.

- Když $m = n$, je matice $\mathbf{A}$ čtvercová.

- Když $m < n$ říkáme, že soustava je (viditelně) *nedourčená*.
- Když $m = 1$ a $n > 1$, hovoříme o jedné lineární rovnici pro více neznámých.

Blok 3.4.12: Základní úloha lineární algebry

Nechť $\mathbf{A}$ je obdélníková matice typu (m, n) , tj. m řádků a n sloupců a $\mathbf{b} \in \mathbb{E}^m$. Soustava (úloha):

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

má řešení v $\mathbb{E}^n$, když:

$$\forall \mathbf{b} \in \mathbb{E}^m \exists \hat{\mathbf{x}} \in \mathbb{E}^n : \mathbf{A}\hat{\mathbf{x}} = \mathbf{b} \text{ (rovnost).}$$

Transponovaná matice k matici $\mathbf{A}$ je matice $\mathbf{B}$ typu (n, m) , která je definována rovností skalárních součinů:

$$\mathbf{z}^T \mathbf{A}\mathbf{x} = \mathbf{x}^T \mathbf{B}\mathbf{z}, \forall \mathbf{x} \in \mathbb{E}^n, \forall \mathbf{z} \in \mathbb{E}^m.$$

Značí se $\mathbf{B} = \mathbf{A}^T$. Všimneme si, že $\mathbf{z}^T \mathbf{A}\mathbf{x}$ je skalární součin v $\mathbb{E}^m$ a $\mathbf{y}^T \mathbf{B}\mathbf{z}$ je skalární součin v $\mathbb{E}^n$.

Mějme lineární zobrazení $\mathbf{A} : \mathbb{E}^n \rightarrow \mathbb{E}^m$ a $\mathbf{A}^T : \mathbb{E}^m \rightarrow \mathbb{E}^n$, značíme: $\mathbf{A} \in \mathbb{E}^{m,n}$, $\mathbf{A}^T \in \mathbb{E}^{n,m}$. Množina:

$$\mathcal{H}(\mathbf{A}) = \{\mathbf{y} \in \mathbb{E}^m : \mathbf{y} = \mathbf{A}\mathbf{x}, \forall \mathbf{x} \in \mathbb{E}^n\} \subset \mathbb{E}^m$$

se nazývá *obor hodnot* lineárního zobrazení určeného maticí $\mathbf{A}$. Je to lineární prostor (prověřte!), jehož bázi jsou *lineárně nezávislé sloupce* matice $\mathbf{A}$ (zápis $\mathbf{A}\mathbf{x} = \mathbf{y}$ říká, že $\mathbf{y} \in \mathbb{E}^m$ je lineární kombinace sloupců matice $\mathbf{A}$).

Množina:

$$\mathcal{H}(\mathbf{A}^T) = \{\mathbf{w} \in \mathbb{E}^n : \mathbf{w} = \mathbf{A}^T \mathbf{z}, \forall \mathbf{z} \in \mathbb{E}^m\} \subset \mathbb{E}^n$$

je obor hodnot lineárního zobrazení určeného maticí $\mathbf{A}^T$. Bázi tohoto lineárního prostoru jsou *lineárně nezávislé řádky* matice $\mathbf{A}$ (prověřte!).

Z teorie lineární algebry víme, že sloupcová hodnota matice $\mathbf{A}$ je rovna řádkové hodnotě, tj. platí:

$$h(\mathbf{A}) = \dim \mathcal{H}(\mathbf{A}) = \dim \mathcal{H}(\mathbf{A}^T) = h(\mathbf{A}^T) \leq \min\{m, n\}.$$

Připomeňme, že $\dim \mathcal{H}(\mathbf{A})$ je počet lineárně nezávislých sloupců matice $\mathbf{A}$ (= počet lineárně nezávislých řádků matice $\mathbf{A}^T$) a $\dim \mathcal{H}(\mathbf{A}^T)$ je počet lineárně nezávislých sloupců matice $\mathbf{A}^T$ (= počet lineárně nezávislých řádků matice $\mathbf{A}$).

Množina:

$$\mathcal{N}(\mathbf{A}) = \{\mathbf{v} \in \mathbb{E}^n : \mathbf{A}\mathbf{v} = \mathbf{0}_m\} \subset \mathbb{E}^n, \mathbf{0}_m \text{ je nulový sloupec v } \mathbb{E}^m,$$

se nazývá *nulový prostor* (jádro) lineárního zobrazení určeného maticí $\mathbf{A} \in \mathbb{E}^{m,n}$. Je to lineární prostor (prověřte!), jehož bázi jsou lineárně nezávislá řešení homogenní soustavy $\mathbf{A}\mathbf{x} = \mathbf{0}_m$.

Množina:

$$\mathcal{N}(\mathbf{A}^T) = \{\mathbf{u} \in \mathbb{E}^m : \mathbf{A}^T \mathbf{u} = \mathbf{0}_n\} \subset \mathbb{E}^m, \quad \mathbf{0}_n \text{ je nulový sloupec v } \mathbb{E}^n,$$

se nazývá *nulový prostor* lineárního zobrazení určeného maticí $\mathbf{A}^T \in \mathbb{E}^{n,m}$. Bází tohoto lineárního prostoru jsou lineárně nezávislá řešení homogenní soustavy $\mathbf{A}^T \mathbf{z} = \mathbf{0}_n$.

Definujeme *ortogonální doplňky* oborů hodnot zobrazení $\mathbf{A}$, resp. $\mathbf{A}^T$:

$$[\mathcal{H}(\mathbf{A})]^\perp = \{\mathbf{z} \in \mathcal{R}^M : \mathbf{z}^T \mathbf{y} = 0; \forall \mathbf{y} \in \mathcal{H}(\mathbf{A})\} \subset \mathcal{R}^M;$$

$$[\mathcal{H}(\mathbf{A}^T)]^\perp = \{\mathbf{v} \in \mathcal{R}^N : \mathbf{v}^T \mathbf{w} = 0; \forall \mathbf{w} \in \mathcal{H}(\mathbf{A}^T)\} \subset \mathcal{R}^N.$$

Věta o řešitelnosti úlohy $\mathbf{Ax} = \mathbf{b}$:

Mějme matici $\mathbf{A}$ s hodnotí $h = h(\mathbf{A})$.

1. Když $\mathbf{b} \in \mathcal{H}(\mathbf{A}) \subset \mathbb{E}^m$ a $h < n$, potom soustava $\mathbf{Ax} = \mathbf{b}$ má v $\mathbb{E}^n$ nekonečně mnoho řešení $\mathbf{x} \in \mathbb{E}^n$ a tato řešení jsou ve tvaru $\mathbf{x} = \bar{\mathbf{x}} + \mathbf{u}$, kde $\mathbf{u} \in \mathcal{N}(\mathbf{A})$ a $\mathbf{A}\bar{\mathbf{x}} = \mathbf{b}$ ($\mathbf{u}$ je libovolný prvek z $\mathcal{N}(\mathbf{A})$); $\bar{\mathbf{x}}$ je nějaký prvek z $\mathbb{E}^n$ splňující rovnost $\mathbf{A}\bar{\mathbf{x}} = \mathbf{b}$, (tzv. *partikulární řešení*).
2. Když $\mathbf{b} \notin \mathcal{H}(\mathbf{A})$, potom soustava $\mathbf{Ax} = \mathbf{b}$ nemá řešení, tj. pro libovolné $\mathbf{x}_0 \in \mathbb{E}^n$ je $\mathbf{Ax}_0 \neq \mathbf{b}$.
3. Když $h = n$ a $\mathbf{b} \in \mathcal{H}(\mathbf{A})$ (speciálně pro $h = n = m$ nemůže být $h > m$), potom soustava $\mathbf{Ax} = \mathbf{b}$ má právě jedno řešení.
4. Soustava $\mathbf{Ax} = \mathbf{b}$ má řešení pro libovolné $\mathbf{b} \in \mathbb{E}^m$ právě tehdy, když $h = m$ (matice $\mathbf{A}$ má plnou hodnot).
5. Následující výroky jsou ekvivalentní:
 - soustava $\mathbf{Ax} = \mathbf{b}$ je řešitelná;
 - $\mathbf{b} \in \mathcal{H}(\mathbf{A})$; Tato podmínka říká, že $\mathbf{b}$ je lineární kombinací sloupců matice $\mathbf{A}$ a odtud se dá pak odvodit známá podmínka řešitelnosti : hodnota matice soustavy se rovná hodnotě matice rozšířené.
 - $\mathbf{z}^T \mathbf{b} = 0, \forall \mathbf{z} \in \mathcal{N}(\mathbf{A}^T)$, tj. pravá strana $\mathbf{b}$ je ortogonální ke všem řešením $\mathbf{z} \in \mathbb{E}^m$ homogenní soustavy $\mathbf{A}^T \mathbf{z} = \mathbf{0}_n$.

Blok 3.4.13: Řešitelnost úlohy $\mathbf{Ax} = \mathbf{b}$

Kapitola 4

Matematické modely některých zákonů

Obsah kapitoly

4.1	Zákony ekonomické rovnováhy	54
4.1.1	Zákon nabídky a poptávky	54
4.1.2	Model uzavřené ekonomiky	58
4.2	Zákony mechanické rovnováhy	60
4.2.1	Archimédův zákon	60
4.2.2	Stavové zákony	61

Motto: Celý Vesmír je založen na principu rovnováhy. Tato rovnováha drží pohromadě nejen Vesmír, ale vše, co se v něm nachází. Včetně jeho základních prvků, atomů...

Homo sapiens byl nucen během svojí existence hledat *rovnováhu* (balance, équilibre, equilibrio, statera, gleichgewicht) mezi svými životními zájmy a *stavem* prostředí, ve kterém žil.

Vymyslel si proto čísla, aby mohl měřit. Objevil úměru, poměr, trojčlenku, aby poznal vztahy a mohl formulovat závěry. Proto vymyslel symbol „=“.

Cílem (nejen) kapitoly 4 je plnění slibu uvedeném v názvu naší série sešitů, a to *Matematické technologie pro modelování*. Nemáme v úmyslu suplovat učebnicový výklad fyziky, či ekonomie.

Chceme na všeobecně známých elementárních přírodních zákonech a ekonomických procesech a jevech ukázat technologii vytváření *matematických modelů* a ukázat proč se tyto modely vytvářejí. Speciálně chceme bojovat proti „vzorečkovému“ pohledu (nejen) na přírodní vědy.

4.1 Zákony ekonomické rovnováhy

Motto: Není to laskavost řezníka, sládka, nebo pekaře, které vděčíme za svůj oběd, ale je to jejich ohled na jejich vlastní zájem. Nespolehneme se na jejich lidskost, ale na jejich sebelásku a nikdy jim nezdůrazňujeme naše potřeby, ale jejich výhody.

Adam Smith

Motto: Když se jedinec ze všech sil snaží použít svůj kapitál na podporu domácího průmyslu tak, aby výroba měla co největší hodnotu, pak nutně usiluje o to, aby roční příjmy společnosti byly co nejvyšší. Ovšemže většinou není jeho úmyslem podporovat veřejný zájem a ani neví, nakolik jej podporuje. Myslí je na svůj zisk a v tom jako v mnohém jiném jej vede *neviditelná ruka*, aby podporoval cíl, který vůbec neměl v úmyslu.

Adam Smith

4.1.1 Zákon nabídky a poptávky

První „krůčky“ matematického modelování spočívají v tom, že si uvědomíme názvosloví, termíny, veličiny a odhadujeme různé závislosti, které jsem schopni interpretovat matematicky.

Trh je místo, kde se střetává nabídka (výrobci a prodejci) s poptávkou (zákazníci, kupující).

Cena (p - price) závisí jednak na čase t , ve kterém dochází ke směně produktu, tak i na místě a podmínkách směny, na prodávajícím a kupujícím. Je to náhodná (stochastická) proměnná. Je to skalární veličina a vyjadřujeme ji v násobcích nějaké cenové jednotky. Tedy $p(t)$, nebo p_t označuje cenu v časovém okamžiku t .

Poptávka (D - demand) je ochota a schopnost kupujících koupit si určité množství daného produktu v závislosti na jeho ceně, tj. D označuje množství poptávaného produktu (počet, tuny, vagony, litry) v závislosti na ceně. Označení $D(t)$, nebo D_t říká, že nás zajímá množství poptávaného produktu v čase t .

Množství (Q - quantum) poptávaného ($\dot{Q}_D$), nebo nabízeného (Q_S) produktu.

Nabídka (S - suggestion) je množství určitého produktu, který prodávající nabídne k prodeji při jeho různých cenách, tj. S je množství nabízeného produktu při určité

ceně. Označení $S(t)$, nebo S_t říká, že nás zajímá množství nabízeného produktu v čase t .

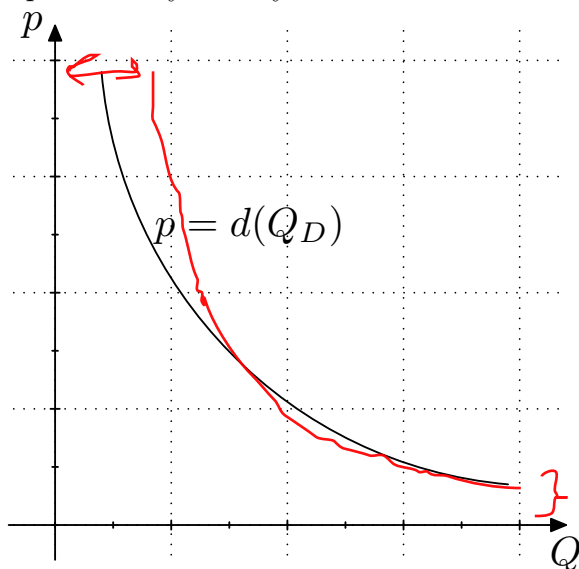
Zákon klesající poptávky:

Poptávka se řídí cenou, která je na trhu, tj. s rostoucí cenou klesá množství poptávaného zboží.

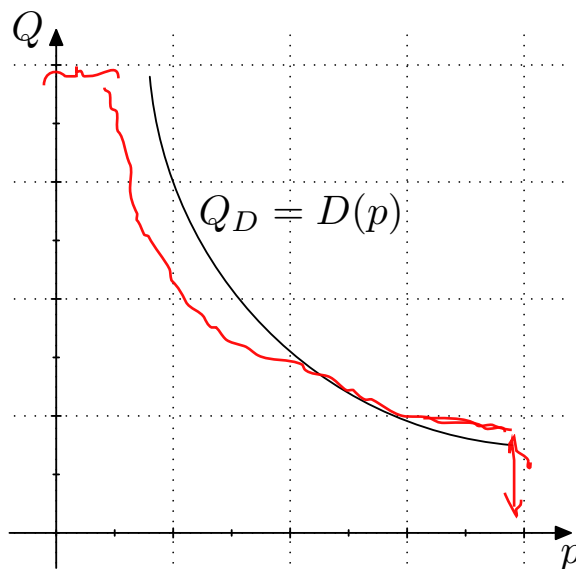
Matematickým modelem tohoto (ideálního) zákona je ostře klesající konvexní hladká funkce $p = d(Q_D)$. její graf se nazývá *poptávková křivka* (viz obrázek 4.1.1).

Lze z ní určit (odhadnout) předpokládané ceny při určitém poptávaném množství produktu na trhu. *Změnu poptávky* znázorňujeme posunem křivky.

Inverzní funkce $Q_D = D(p)$ je opět ostře klesající konvexní a hladká. Vypovídá o tom, že při rostoucí ceně poptávané množství Q_D klesá, tj. zákazník je méně ochoten nakupovat za vyšší ceny.



Obrázek 4.1.1: Zákon $p = d(Q_D)$



Obrázek 4.1.2: Zákon $Q_D = D(p)$

Poznámka: V učebnicích matematiky se uvedené vztahy zapisují:

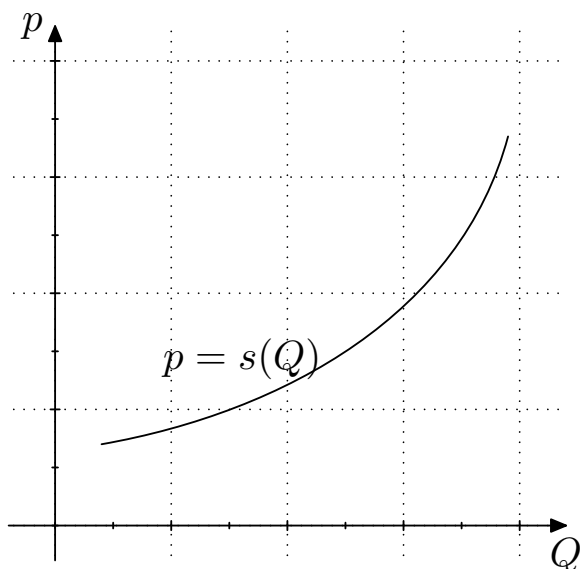
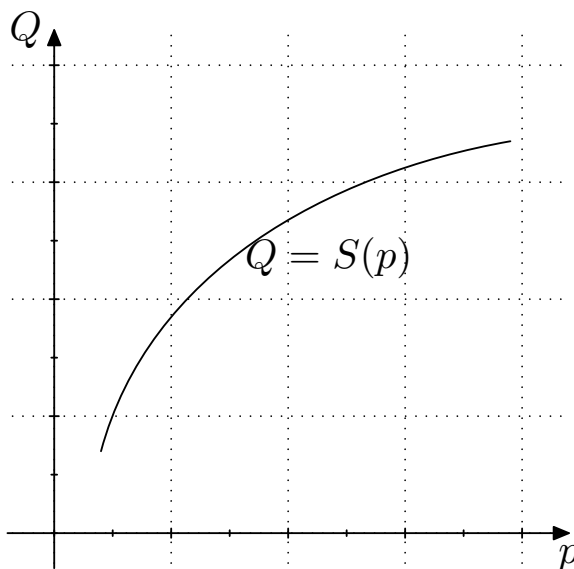
$$y = f(x) \leftrightarrow x = f^{-1}(y)$$

Zákon rostoucí nabídky:

Růst ceny vyvolává růst nabídky, neboť zvyšuje zájem ze strany výrobců.

Matematickým modelem tohoto (ideálního) zákona je ostře rostoucí konvexní hladká funkce $p = s(Q_S)$, jejíž graf se nazývá *nabídková křivka* (viz obrázek 4.1.3). Lze z ní určit (odhadnout) předpokládané ceny při určitém nabízeném množství produktu na trhu. *Změnu nabídky* opět znázorňujeme posunem křivky.

Inverzní funkce $Q_S = S(p)$ je opět ostře rostoucí, hladká, ale konkávní. Vypovídá o tom, že při rostoucí ceně nabízené množství Q_S roste, tj. výrobce dodává na trh větší množství produktů (větší objem produkce) - viz obrázek 4.1.4.

Obrázek 4.1.3: $p = s(Q_S)$ Obrázek 4.1.4: $Q = S(p)$

Q_S je nabízené množství.

Zákon rovnováhy nabídky a poptávky:

- rovnováha na trhu jednoho produktu je taková situace, kdy nabízení množství Q_S se rovná poptávanému množství Q_D
- rovnováha na trhu jednoho produktu je taková situace, kdy cena nabídky p_S se rovná ceně poptávky p_D , tj. $d(Q_D) = s(Q_S)$.

Matematickým modelem zákona je rovnost:

$$d(Q_D) = s(Q_S) \text{ nebo } D(p_D) = S(p_S) \text{ nebo } (Q_D)_t = (Q_S)_t,$$

která říká, že v určitém čase t bude poptávané množství rovno nabízenému množství.

Matematické modely zákonů se formulují především proto, aby bylo možné formulovat *matematické úlohy* pro určení parametrů vystupujících v modelu.

Úlohy rovnováhy:

1. Pro dané poptávkové a nabídkové křivky (funkce) stanovit *rovnovážnou cenu* p^* na trhu takovou, že platí rovnost:

$$D(p^*) = S(p^*)$$

Máme tedy řešit rovnici $D(p) - S(p) = 0$, pro jednu neznámou (viz odstavec 3.1.4).

2. Pro dané poptávkové a nabídkové křivky (funkce) stanovit *rovnovážné množství* Q^* produktů takové, že platí rovnost:

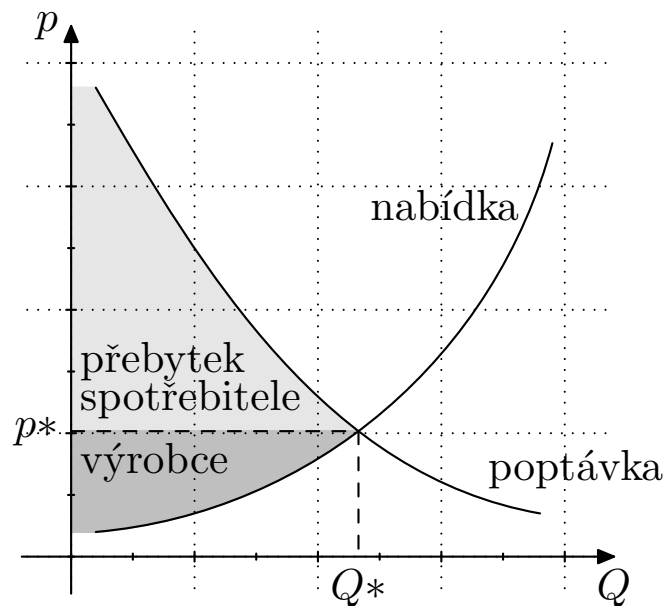
$$d(Q^*) = s(Q^*).$$

Máme opět řešit rovnici, tentokrát $d(Q) - s(Q) = 0$, pro neznámou Q .

Je evidentní, že největší problém těchto úloh jsou vstupní data. Na obrázcích to vypadá jednoduše, má se stanovit průsečík uvedených křivek. ovšem matematické řeči o řešitelnosti rovnic zde již mají konkrétní obsah.

Například, může nastat na popisovaném trhu rovnováha? Existuje vůbec (na reálném trhu) rovnovážná cena?

Idealizované (fiktivní) modely jsou však velmi užitečné, protože umožňují klást otázky, jejich odpovědi již mají praktický obsah. Jedna taková situace je na obrázku 4.1.5.



Obrázek 4.1.5: Rovnováha nabídky a poptávky

→ **Dynamický zákon nabídky a poptávky:** jež zformulujeme v podobě matematického modelu, tj. jako rovnost:

$$D(p_t) = S(p_{t-1}).$$

Touto rovností je definovaná posloupnost cen $\{p_t\}$, $t = 1, 2, 3, \dots$; v čase t . Zde p_t představuje aktuální cenu (cenu na aktuálním trhu) v čase t a p_{t-1} představuje cenu z předchozího období.

Princip metody prosté iterace (viz odstavec 3.1.2) pro řešení rovnic, nám umožní stanovit vývoj cen v jednotlivých fázích konkurenčního trhu. Musíme však zvolit p_0 .

Graficky znázorněné jednotlivé mezivýsledky iteračního procesu se (k údivu matematiků) nazývají *pavučinový model zákona nabídky a poptávky* (viz obrázek 4.1.6).

Posloupnost odchylek od rovnovážné ceny $p_t - p^*$ umožňuje analyzovat proces konvergence k rovnovážné ceně.

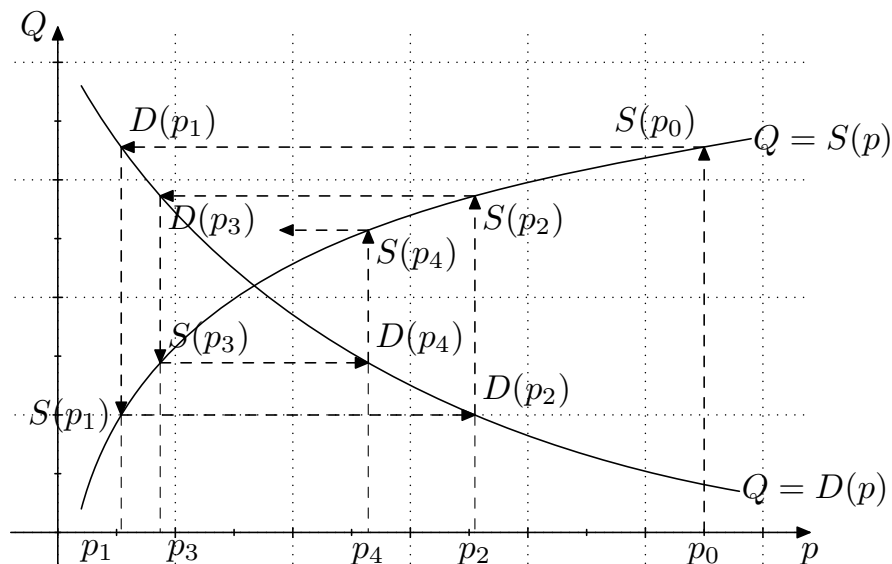
Jednotlivé iterační kroky:

1. zvolíme p_0
2. $S(p_0) = D(p_1) \Rightarrow p_1 :$
3. $S(p_1) = D(p_2) \Rightarrow p_2 :$

$$4. S(p_2) = D(p_3) \Rightarrow p_3 :$$

$$5. S(p_3) = D(p_4) \Rightarrow p_4 :$$

6. ...



Obrázek 4.1.6: Pavučinový model zákona nabídky a poptávky

4.1.2 Model uzavřené ekonomiky

Jedinou uzavřenou ekonomikou je fakticky globální ekonomika, neboť s jinými světy (zatím) si nevyměňujeme komodity. Jsou ovšem regionální oblasti (dokonce i státy), které mají určité znaky uzavřené ekonomiky.

Chceme vysvětlit princip ekonomické rovnováhy (stability) za předpokladu, že uvažovaný ekonomický systém je *uzavřený*, tzn. ekonomická rovnováha vychází z „přirozeného“, tj. idealizovaného pohledu na ekonomické vztahy.

Tento pohled implicitně definuje jistý stupeň sociální spravedlnosti. Nikdo ze zúčastněných „aktérů“ nebude doplácet na ostatní a nikdo se na úkor ostatních neobohatí. Jedním z cílů je najít parametry rovnováhy, tj. ceny (mzdy) v jednotkách, které si stanovíme, například „počet medvědíků kožešin“ určité velikosti - jednotka M_k .

Zákon rovnováhy formulujeme takto:

V prostředí uzavřeného systému výrobců, poskytovatelů spotřebitelů služeb rovnováhou rozumíme takový stav, v němž celkové náklady C_N se rovnají celkovým příjmům C_P , tj. platí rovnost:

$$C_N = C_P.$$

Příklad užití zákona rovnováhy v prostředí „chalupářské ekonomiky“:

Nechť uzavřený systém je tvořen třemi poskytovateli služeb: *Tesařem, Elektrikářem a Instalátérem*.

Navzájem se dohodli, že si udělají opravy na svých chalupách. Po zhodnocení stavu svých chalup rozhodli o objemu prací (cena práce bude tvořit mzdu, v jednotkách M_k) a že to udělají za 10 dnů a zaplatí si navzájem svojí prací. Sepsali si pracovní a výplatní plán

Odpracováno pro	Objem prací odvedených (počet pracovních dnů)		
	Tesařem	Elektrikářem	Instalatérem
Tesaře	2	1	6
Elektrikáře	4	5	1
Instalatéra	4	4	3

Označíme-li p_i jednotkový (denní) objem práce vyjádřený mzdou, pak

p_1 je denní mzda Tesaře,

p_2 je denní mzda Elektrikáře,

p_3 je denní mzda Instalatéra.

Potom podle dohody respektující princip rovnováhy musí být

náklady tesaře: $2p_1 + p_2 + 6p_3 = 10p_1$ (příjem tesaře),

náklady elektrikáře: $4p_1 + 5p_2 + p_3 = 10p_2$ (příjem elektrikáře),

náklady instalatéra: $4p_1 + 4p_2 + 3p_3 = 10p_3$ (příjem instalatéra).

Tato soustava podmínek (rovností) tedy představuje matematický model zákona (principu) ekonomické rovnováhy v ekonomickém systému tří chalupářů vyjádřeného pouze pomocí mezd a objemu prací. Potřebný materiál (jeho cenu) zahrnujeme do ceny práce.

Můžeme formulovat úlohu rovnováhy (soustavu tří lineárních homogenních rovnic pro hledání neznámých p_1, p_2, p_3). Shodou „šťastných“ dohod chalupářů je soustava řešitelná (matice soustavy je singulární).

Lze snadno stanovit výsledek, řešení této homogenní soustavy. Je to jednoparametrický systém vektorů

$$\mathbf{p} = \beta [31, 32, 36]^T,$$

kde parametr β je libovolné reálné číslo.

Interpretace:

Ekonomika tří chalupářů bude v rovnováze pouze tehdy, budou-li jejich mzdy určeny vypočteným vektorem $\mathbf{p}$.

Například ($\beta = \frac{1}{10}$): $p_1 = 0,31M_k$ $p_2 = 0,32M_k$ $p_3 = 0,36M_k$

Poznámka: pro jiný technický stav chalup (jiný plán) budou platit jiné ekonomické podmínky – jiná dohoda.

Obecná úloha

Mějme systém n poskytovatelů vzájemných služeb (výrobců). Každý z nich poskytuje zboží, služby pro ostatní výrobce a užívá dohodnutým (určeným) způsobem zboží, služby ostatních výrobců. Chceme stanovit ceny zboží, služeb tak, aby celkové výlohy byly rovny celkovým příjmům.

Označme:

p_j : cena (objem) práce j -tého výrobce za zvolený časový interval,
 a_{ij} : objem práce vyjádřený počtem časových jednotek práce v počtu γ (část celkových výloh) j -tého výrobce pro i -tého výrobce.

Předpokládáme:

$p_j \geq 0$: $i = 1, 2, \dots, n$,
 $a_{ij} \geq 0$: $i, j = 1, 2, \dots, n$,
 $\sum_{i=1}^n a_{ij} = \gamma$: výlohy všech výrobců pro j -tého výrobce,
 $\mathbf{p} = [p_1, p_2, \dots, p_n]^T$: cenový vektor,
 $\mathbf{A} = (a_{ij})$: burzovní (peněžní, kurzovní) matice,
 γ : parametr objemu práce.

Matematický model zákona ekonomické rovnováhy je dán rovností

$$\mathbf{A}\mathbf{p} = \gamma\mathbf{p} \quad , \text{ resp. } \sum_{j=1}^n a_{ij}p_j = \gamma p_i \quad , i = 1, 2, \dots, n.$$

Matematická úloha je rovnice pro neznámé $\mathbf{p}, \gamma$, příp. a_{ij} . Chceme stanovit takové hodnoty $\hat{\mathbf{p}}, \hat{\gamma}$, pro které nastane rovnováha.

Poznámka: Podmínkou rovnováhy je tedy řešitelnost formulované úlohy. Pokud některý z poskytovatelů požaduje vyšší mzdu, ostatní nemají čím platit a rovnováha nenastane.

4.2 Zákony mechanické rovnováhy

4.2.1 Archimédův zákon

Každý potenciální sebevrah, který se rozhodne odejít ze světa utopením by si měl uvědomit detaily Archimédova zákona:

těleso ponořené do kapaliny je nadlehčováno vzlakovou silou, jejíž velikost se rovná tíze kapaliny stejného objemu, jako je objem ponořeného tělesa.

V této slovní formulaci však nejsou informace, za jakých podmínek bude pokus sebevraha úspěšný. V tekutém medu bude jeho pokus neúspěšný, v těžké vodě, či mořské vodě může být pokus méně úspěšný. Naopak, ve vařící vodě bude pokus určitě úspěšný.

To vše závisí na veličině, které se říká *hustota* (kapaliny, plynu, pevné látky). Je to *skalár* a vyjadřuje „kolik kilogramů látky se vejde do jednoho krychlového metru“. Značí se ϱ a závisí na teplotě, hmotnosti, tlaku.

Vzlaková (nadlehčující) síla v zemském gravitačním poli je:

$$\mathbf{F}_{vz} = \varrho_{kap} \cdot V_{kap} \cdot \mathbf{g},$$

kde $\mathbf{g}$ je gravitační zrychlení a součin $\varrho_{kap} \cdot V_{kap}$ je hmotnost kapaliny stejného objemu, jako je objem ponořené části tělesa.

Je-li $G_{těl} = \rho_{těl} \cdot V_{těl} \cdot g$ tíha tělesa mimo kapalinu a G_{kap} tíha téhož tělesa ponořeného v kapalině, potom rovnost:

$$G_{kap} = G_{těl} - F_{vz},$$

resp. rovnost:

$$G_{kap} = \rho_{těl} \cdot V_{těl} \cdot g - \rho_{kap} \cdot V_{kap} \cdot g$$

je *matematickým modelem Archimédova zákona*. Jde o podmínku *rovnováhy sil*.

Tato rovnost se stane rovnicí, když určíme, která z veličin je hledaná. Význam matematického (byť jednoduchého, až primitivního) modelu spočívá v tom, že nám dovoluje formulovat další otázky a pomocí takto formulovaných úloh hledat odpovědi:

1. Popsat pohyb těles (ponorky, lodě, plavce) v kapalinách s proměnnou hustotou, teplotou, tlakem; plavec svým pohybem zvětšuje vztlakovou sílu. Pohyb plavce se kromě Archimédova zákona řídí také zákonem pohybovým (*okamžitá změna hybnosti je určena časově proměnnou silou*).

Pak je potřeba řešit úlohu, kterou je soustava rovnic. Jedna algebraická a druhá diferenciální. V aplikacích se takovým úlohám říká *Algebraicko-diferenciální rovnice*.

2. Rodiče s dítětem v bazénu řeší (intuitivně) úlohu najít takovou hustotu soustavy dítě-plovací vesta, aby $G_{kap} = 0$ (hustota lidského těla je asi $1166 \text{ kg} \cdot \text{m}^{-3}$, hustota vody $960 - 1000 \text{ kg} \cdot \text{m}^{-3}$).
3. Rozšířit platnost zákona i na plyny (balony) a jaká jsou omezení na jeho platnost.
4. Jestliže na ponořené těleso působí kromě gravitační síly $G_{těl}$ ještě nějaká jiná tažná síla (například tažná síla žraloka držícího plavce za nohu), není problém matematický model modifikovat. Dá se pak vypočítat, že naděje na rovnováhu je mizivá.

Dá se říci, že Archimédes položil základy *hydromechaniky*.

4.2.2 Stavové zákony

Motto: Energie, práce, teplo ($\text{joule} = \text{kg} \cdot \text{m}^2 \cdot \text{s}^{-2}$) není totéž; pojem energie je spojen se stavem, kdežto pojmy teplo a práce se vztahují k dějům.

Oblast fyziky, resp. termomechaniky, termodynamiky, v nichž se pojednává a *stavových zákonech* má z pohledu studenta největší „výskyt vzorečků“ na čtvereční metr textu.

Použijeme-li naší koncepci matematického modelování, jsme přesvědčení, že obrovský rozsah poznatků se stane srozumitelnější.

Na Wikipedii jsme našli tuto formulaci:

Dnešní podoba stavové rovnice vznikla postupnou syntézou (vzájemným zobecněním) tří parciálních zákonů pro tři izo-procesy. Izotermicky, izobaricky a izochoricky děj popsany Boyle-Mariottovým, Gay-Lussacovým a Charlesovým zákonem spolu s přibráním Avogadrova zákona, univerzální plynové konstanty a dospění k pojmu „absolutní termodynamické teploty“

Připadá nám podivné, že nějaká rovnice je zobecněním nějakého zákona. Někteří autoři si tento spíše jazykový nesmysl uvědomují a místo termínu „fyzikální zákon“ nabízejí termín „veličinová rovnice“.

Matematický jazyk pěstujeme také proto, aby nabídl srozumitelnější překlad jazyka přírody. Doporučujeme toto pořadí:

1. **Procesy a jevy** v přírodě - objekty našeho zájmu.
2. **Modely procesů a jevů** v odborném jazyce - většinou slovně pomocí popisů a obrázků určovat závislosti, příčiny a následky.

Například *matematické kyvadlo* (hmotný bod na nehmotné niti) je fyzikální model pohybu zavěšeného tělesa.

3. **Matematický model** - pomocí matematických prostředků formulování (idealizovaných) vztahů (obvykle rovností) mezi definovanými veličinami. Těmto vztahům říkáme *modely zákonů* (někteří autoři používají termín *matematický zápis zákona*).

V praxi (nejen školní) se zákon ztotožní s jeho matematickým modelem a navíc se mu přisoudí název *rovnice*. Příkladem je právě termín *stavová rovnice*. Nelíbí se nám to, ale můžeme proti tomu jen protestovat.

4. **Formulace úlohy** (obvykle rovnice) pro hledání parametru procesu, pro která jsme model vytvářeli.
5. **hledání řešení** formulované úlohy a srovnání s realitou.

Jestliže empirické poznatky domyslíme a zobecníme do podoby modelů a úloh, můžeme na základě matematických poznatků a technologií dospět k novým výsledkům, které vypovídají o skutečnosti.

Bez matematiky by poznání přírody bylo pouhým popisem pozorovaných jevů.

Stavový zákon formuluje vztah stavových veličin (parametrů, veličin kvality)

Vnitřní:

p ... tlak, $[p] = \text{kg m}^{-1} \text{s}^{-2}$, $\text{Pa} = \text{Nm}^{-2}$ (pascal)

T ... termodynamická teplota, stupeň Kelvina

ρ ... hustota, $[\rho] = \text{kg m}^{-3}$

Vnější:

V ... objem, $[V] = m^3$

m ... hmotnost, $[m] = kg$

n ... látkové množství, $Mol = 6 \cdot 10^{23}$ částic

T ... termodynamická teplota, stupeň Kelvina

R ... molární plynová konstanta, $R = 8,3144621 J \cdot mol \cdot K^{-1}$

k ... počet částic

S ... entropie, $[S] = J \cdot K^{-1}$ (Joule/kelvin)

Q ... teplo (vnitřní energie), $[Q] = kg m^2 s^{-2}$, Joule

Zkušenosti a fyzikální experimenty vedly k závěrům (hypotézám), že stavové parametry, především *tlak*, *teplota*, *objem*, nejsou nezávislé a že tedy musí platit *vazby* typu:

$$F(p, V, T, \rho) = 0.$$

Zkoumání těchto vazeb je spojováno především se jmény:

Robert Boyle (1660), Robert Hook (1667), Edme Mariotte (1679), Joseph Louis Gay-Lussac (1802), Jacques Alexandre César Charles (1787), Johan Diderik van der Waals (1900).

Postupně se dospělo k závěru, v plynech jsou stavové parametry vázány algebraickou rovností:

$$pV = nRT,$$

kde n je látkové množství plynu, R je molární plynová konstanta a T je absolutní (termodynamická) teplota plynu.

Postupně se však zjišťovalo, že v žádném konkrétním plynu tato rovnost není splněna přesně a že plyn, ve kterém platí přesně vlastně neexistuje.

Tomuto neexistujícímu plynu se dal název *ideální plyn* a uvedenému vztahu se začalo říkat *stavový zákon ideálního plynu*. Nejblíže k ideálnímu plynu má prý přehřátá pára.

Proces zpřesňování popisu chování plynu dále pokračoval. Uvedme pouze matematické modely příslušných zákonů:

Van der Waalsův zákon:

$$\left(p + n^2 \frac{a}{V^2}\right) (V - nb) = nRT$$

Berthelotův zákon:

$$\left(p + n^2 \frac{a'}{TV^2}\right) (V - nb) = nRT$$

Clausiův zákon:

$$\left(p + n^2 \frac{a'}{T(V + nb')^2}\right) (V - nb) = nTZ$$

Můžeme prezentovat další matematické modely termodynamických zákonů rovnováhy, ovšem už by nám jenom algebraické prostředky nestačily. Existuje velké množství zdařilých učebnic. Velmi zasvěcený se nám zdá text: *J. Novák a kolektiv: Fyzikální chemie, VŠCHT 2008*.

Na závěr si dovoluujeme (ta troufalost!) předložit následující úkoly: Ze seriózních zdrojů vyhledat slovní formulace obecných termodynamických zákonů a přiřadit jim zápisy jejich matematických modelů:

1. princip růstu entropie (2. věta termodynamická, Clausiova rovnice),

2. Nernstův teorém (3. věta termodynamická),
3. zákon zachování energie (1. věta termodynamická).

Kapitola 5

Hovory M

Obsah kapitoly

5.1 Školské formulace úloh	65
5.2 Eliminační principy	66
5.2.1 Podivuhodná role expertů na základní škole	66
5.2.2 Determinanty a další starožitnosti	67
5.3 Opět Al-Chwárizmí	67
5.3.1 Historie evropské (nejen matematické) vzdělanosti	67
5.3.2 Díla Al-Chwárizmího	69
5.4 Celý svět modeluje	70
5.4.1 Doby stárnou rychleji než lidé	73
5.4.2 Matematické modelování a vzdělávání modelářů	73
5.4.3 „Antimodelářská“ výuka matematiky	76
5.5 Podrobnosti o slavných úlohách	76
5.5.1 Pověst o založení Kartága	76
5.5.2 Brachystochrona nebo brachistochrona	78

5.1 Školské formulace úloh

Silně se přimlouváme za to, aby se ve školní praxi nepoužívaly příliš stručné, případně nepřesné, nejednoznačné formulace tréninkových úkolů typu:

- Upravte (zjednodušte, „ne-zjednodušte“) algebraický výraz: ...
(obvykle se předpokládá, že podrobné vstupní informace jsou dány výkladem učitele a že jsou některé úpravy vysloveně zakázány. Jinak by stačilo zaměnit výraz typu $a + b$, výrazem $b + a$, případně výrazem $a - b$, a úkol by byl splněn, našel se upravený výraz.)
- Řešte kvadratickou rovnici ...
(pokud „školnímu“ řešiteli není zřejmé, že si musí ujasnit vstupní informace a výstupní požadavky a že metoda „vzorečku“ je důsledkem metody převodu na odmocňování).

- Sestrojte trojúhelník ze tří výšek, rozdělte úhel na tři stejné části, ...
Na druhé straně v ne-školní praxi musí matematik počítat s tím, že již převod technologického problému na matematickou úlohu (sérii matematických úloh) bývá samostatným složitým procesem.

5.2 Eliminační principy

5.2.1 Podivuhodná role expertů na základní škole

Přesto si dovolíme vyprávět příběh „humorný až k pláči“. Na jedné základní škole jeden kvalifikovaný pedagog vyložil problematiku soustavy dvou lineárních rovnic pro dvě neznámé. Jako metodu dal žákům návod:

- z druhé rovnice vypočtete y a dosadíte do první rovnice a vypočtete x ;
- poté dopočítáte číselnou hodnotu y .

Na několika příkladech postup natrénoval. Potom zadal domácí úkol z učebnice. Jedna pečlivá dívka si doma s jedním příkladem nevěděla rady. Byla to soustava:

$$\begin{aligned} 3x + 2y &= 1 \\ 2x &= 4 \end{aligned}$$

Stalo se, že ani rodič neporadil. Naštěstí vedle bydlel vysokoškolský profesor-fyzik. Poradil, i když s velkou nedůvěrou dívky. Na příkaz rodičů nakonec radu akceptovala. Druhý den přišla ze školy s pláčem. Dostala z příkladu pětku, ostatní žáci jedničku, protože pochopili, že to nejde vypočítat (neměli doma konzultanta). Pedagog je pochválil a potvrdil, že opravdu daná soustava nemá řešení!

Zde by příběh mohl končit, ale nekončí. Profesor-fyzik pocítil v dalších dnech značné ochladnutí sousedských vztahů. Nedalo mu to a zeptal se.

Byla mu sdělena krutá skutečnost: poradil blbě a navíc způsobil dívence silné trauma. Všichni ve třídě to měli dobře, jenom ona jediná, premiantka k tomu, to měla špatně.

Profesor-fyzik, spolu s kolegou, profesorem-matematikem, provedli analýzu argumentu (graduovaného) pedagoga o neřešitelnosti dané soustavy. Pravděpodobný závěr byl téměř neuvěřitelný, v souboru výsledků učebnice nebylo u tohoto příkladu nic uvedeno (nebyl tam výsledek!), tak proto je soustava neřešitelná.

Od té doby dívka v dalším studiu nechtěla s matematikou mít nic společného a zůstala jí v duši hluboká averze k matematice. Ke škodě dívky a možná i ke škodě matematiky.

5.2.2 Determinanty a další starožitnosti

5.3 Opět Al-Chwárizmí

Postava Al-Chwárizmího (a její vliv na moderní vývoj matematiky) je natolik významná v „běhu matematických věků“ (a dějinách vzdělanosti vůbec), že jsme usoudili že by bylo velmi nespravedlivé¹ se alespoň letmo nezmínit o některých historických souvislostech spojených s jeho osobou.

Kompilací velkého množství různých zdrojů se pokusíme předložit pravděpodobnou verzi klíčového období vývoje matematiky.

5.3.1 Historie evropské (nejen matematické) vzdělanosti

Již v nejstarších civilizacích byly úlohy řešeny pomocí slovně popsáných postupů a návodů. Tyto postupy se pak zapisovaly na tehdy dostupné materiály, včetně „zápisu“ do paměti. Jen pro sčítání a násobení čísel bylo v průběhu historie vymyšleno spoustu technik (technologií). Především pro počítání s velkými čísly.

Alexandr Veliký prý měl stotisícovou armádu. Kdyby spotřeboval mnoho času na počítání živých a mrtvých vojáků, nezbyl by mu čas na válčení. V Indii a v Číně měli vojáků ještě více.

Vymýšlely se proto různé počítací nástroje:

- Oblázky (lat. Calculi), kuličky (počítadlo s kuličkami = abakus),
- později logaritmické pravítko,
- počítač (comput(e)-er = ten (to), kdo (co) počítá),
- ap.

Učenci od dávných dob zkoumali to, co bylo pro život důležité. Vodu, vzduch, slunce, obživu, majetek a samozřejmě nástroje jak si to opatřit, případně udržet.

Počítání vojáků je jen zlomek toho, co člověk k životu potřeboval. Boj o kořist je základem veškerého lidského konání. Citát z LN 6.3.2015:

*Chamtivost je správná a funguje. Zachycuje esenci evolučního ducha.
Ve všech svých formách nás provázela vzestupem lidstva. Už u zrodu
Archimédova zákona byla chamtivost.*

Dnes se můžeme dohadovat zda *Archimédes* byl matematik, fyzik, filosof, projektant, zbrojař (vyráběl zbraně pro obránce města). Stejně tak ostatní slavní učenci byli „multioboroví“. V té době nikoho nenapadlo tvrdit, že vzdělání a úsilí o poznávání je nepotřebné. Operace sčítání, odečítání, násobení, dělení a další „umění matematické“, především měření délek, obsahů, objemů, úhlů a vytváření geometrických představ se staly fundamentem vzdělanosti.

¹a nemoudré

Nejen *Eukleidovy základy* již obsahují návody a postupy, kterým dnes říkáme *algoritmy*. S ohledem na obecnost algoritmu, jako popisu libovolné běžné každodenní činnosti, sahá počátek vzniku algoritmu až ke vzniku lidstva samotného.

Vždyť i činnosti pravěkých lidí měly přesně daný postup, řád, konečnost a výsledek. Postupy, které dnes splňují podmínky algoritmu, se ve vědních oborech objevily již v období starého Řecka, ačkoli v této době termín „algoritmus“ ještě nebyl vůbec znám.

Ten se v matematice začal používat až v období renesance. Podstatu a použití algoritmu v období cca 300 let př.n.l. dokládá např. známý *Euklidův algoritmus* pro výpočet NSD, (největšího společného dělitele), nebo v 1. století n.l. *Herón z Alexandrie* který algoritmicky podal důkaz vzorce pro obsah trojúhelníku, či algoritmus pro hledání odmocniny z čísla.

Již v Sešitu I jsme se zmínili, že vzhledem k rozpadu Římské říše byli Arabové národem, který zachoval díla starověkých učenců. Pozor! Ještě před rozvojem islámu! V Byzanci, východní části římského císařství, vydával císař Justinian I. (527-565) stále ostřejší výnosy proti vyznavačům různých kacířství a v roce 529 vydal zákaz vyučovat v Athénách jakékoli filozofii.

V právním kodexu, který Justinian uveřejnil čteme odstavec:

- „De maleficiis mathematicis et caeteris similibus“ (O zločincích, matematicích a jiných jim podobných)

a v něm se nachází tento bod:

- „Ars autem mathematica damnabilis interdicta est omnino“ (Zavrženíhodné umění matematické jest zakázáno především).

V té době byly v úpadku všechny světské nauky a s nimi i matematika.

Do 8. století n.l. Arabové ovládli Malou Asii, Kavkaz, Střední Asii, Severní Afriku a Pyrenejský poloostrov. Ke konci 8. století se mnoho evropských, ale i indických či babylonských učenců soustředí do Bagdádu. Právě na znalostech těchto zahraničních učenců se tvoří základy a další rozvoj arabské vědy.

V Bagdádu se zakládá velká knihovna obsahující opisy děl až z Byzance, kam byla pod záštitou chalífy poslána skupina písařů. V době největšího rozkvětu obsahovala okolo 400 000 svazků. Ve městě také existovala celá řada menších knihoven a knižních obchodů. Zatímco Evropa prožívala svoji dobu temna, arabský svět prohluboval vědomosti ve všech rovinách. Nejinak tomu bylo i u matematiky. Je ironií, že mnoho děl se dostalo do arabské říše po překladu z řečtiny či latiny a znovu se do Evropy vrací jako překlady z arabštiny do latiny.

Za panování *Hárúna Ar-Rašída* (asi 17.března 763 až 24.března 809) i jeho syna *Al-Ma'múna* (813 až 833) z Abbásovské dynastie byla vědecká díla starověkých klasiků překládána do arabského jazyka. V Bagdádu byl zřízen *Dům moudrosti*, tj. jakási tehdejší „akademie věd“ s astronomickou observatoří.

Matematici, kteří zde pracovali, měli možnost nejen navázat na dědictví řecké matematiky, ale poznali i matematiku obsazených oblastí (Egypt, Mezopotámie) a matematiku Indie. Arabští učenci tak sehráli důležitou úlohu, neboť uchovali starší poznatky v době kulturního úpadku tehdejší Evropy a rozmnožili je. Správcem byl ustanoven významný perský matematik, astronom a geograf jménem:

- *Abú Abd Alláh Muhammad Ibn Músá al-Chórezmí Abú Dža'far*

Existuje velké množství transkripcí tohoto arabského jména:

- Abū Abd Allāh Muhammad ibn Mūsā al-Khwārizmí,
- Muhammad ibn Musa al-Chvárizmí,
- al-Chvárizmí,
- Muhammed Ibn Músa,
- Abu Jafar Muhammada ibn Mūsā al-Chwārizmí,
- Abú Abd Alláh Muhammad ibn Músá al-Chwārizmí,
- latinizované na Al-Gorizmí, později na Algoritmí.

Nevíme, která transkripce je nejpřesnější. Do češtiny bychom jméno přeložili jako *Otec Abdulláha, Mohameda, syn Mojžíšův, pocházející z Chwārizmu* (Chórezm).

Al-Chwārizmí pocházel pravděpodobně z tehdy perského města Chórezm (dnes Chiva v Uzbekistánu), jeho jméno znamená v arabštině z Chórezmu. Některé zdroje uvádějí jako místo narození Bagdád. Jeho rodným jazykem byla pravděpodobně perština. Svá díla však psal v arabštině, která byla tehdy vědeckým jazykem islámského světa.

Chiva je bývalým hlavním městem říše Chórezm. Na začátku 17. století se Chiva stala hlavním městem Chivského chanátu. 28. května 1873 dobyla Chivu ruská vojska. Po bolševickém převratu se na území bývalého Chivského chanátu ustavila Chórezmská lidová sovětská republika, která se stala v roce 1924 součástí Sovětského svazu.

5.3.2 Díla Al-Chwārizmího

Al-Chwārizmí je autorem traktátu:

Kitáb al-džám'a wa-l-tafríq bil-hisáb al-hindi

asi z roku 825. (Kniha o sčítání a násobení podle indického počtu). V roce 1145 byl přeložen do latiny jako:

Algorithmi de numero indorum.

Prostřednictvím této knihy se evropská věda seznámila s indickou poziční soustavou výpočtů a s číslem 0 (nula).

Je třeba říci, že někteří překladatelé části textu vynechávali, když jej neuměli přeložit. Latinská zkomolenina jména *Al-Gorizmí* se stala základem termínu *algorithmus*. Indické symboly pro číslice (sanskrt) se dostaly do Evropy jako tzv. arabské číslice. Původní text traktátu v arabštině se nezachoval. V češtině jej nazýváme Aritmetický traktát.

Je zajímavé, že při překladu zřejmě došlo k záměně arabského kořene s kořenem řeckého slova:

$\alpha\rho\theta\mu\omicron\varsigma$ (arithmos)

se z algorismu stal algorithmus. (Později bylo v některých jazycích včetně češtiny „th“ změněno na „t“, v katalánštině se vrátilo s.) Toto slovo se používalo jako označení různých matematických postupů, např. v 18. století označoval latinský termín *algorithmus infinitesimalis* „metodu výpočtů s využitím nekonečně malých veličin, vynalezenou Leibnizem“. Slovo algorithmus v dnešním významu se používá až zhruba od 20. století.

Ve svém dalším traktátu:

Al-džabr ua'l-muqábala, resp. Kitáb al-jabr waāl-muqábala

popisuje metodu k vyjádření neznámé (aš-šái, doslova věc) v rovnici prvního stupně (lineární rovnici). Na konci označuje věci (aš-šái) symbolem X.

Jiný zdroj uvádí název traktátu podrobněji:

Al-kitab al-muchtasar fí hisáb al-džabr wa-l-muqábala

(Krátká kniha o počtu připočítávání a porovnávání). Algebraická část traktátu je věnována řešení lineárních a kvadratických rovnic s celočíselnými koeficienty.

Operace „al-džabr“ (algebra) odstraňuje z rovnice člen se záporným znaménkem tím, že se k oběma stranám rovnice odpovídající člen připočítá. Ze slov označujících tuto operaci vzniklo dnešní slovo *algebra*.

V češtině užíváme název *Algebraický traktát*. Je jistě také zajímavé, že Al-Chvárizmí je „vinen“ tím, že v matematice máme termíny *aritmetika*, *algebra*, *algorithmus*.

Poznámka: Citace z Ottova slovníku naučného (1888–1909, určitou dobu byl editorem T.G.M.):

→ Al-Chovárizmí Abú Džafar Muhammed ibn Músá, perský matematik IX.stol. Žil u dvora chalífy Ma'múna, vyniká jako matematik a astronom.

→ Spisy jeho obou oborů, mimo jiné Arithmetika a Algebra, dále proslulé v orientu Tabulky astronomické, vynikaly vhodností metody i jasným podáním daleko nad díla soudobá i pronikly v překladech Adelharda de Bath a Gerarda z Cremony záhy (v XI. stol.) i do Evropy.

→ Arithmetika Clho tvoří první číslo v „Trattati d'arithmeticé“ prince Bon Compagniho, algebru vydal Rosen (Londýn, 1831, text arab. s lat. prekl.). Spisům Clho, v nichž spojuje autor tradici řeckou s názory přejatými z Indie a Persie, náleží veliký vliv na vývoj matematiky v Evropě. Mimo jiné mu děkuje Evropa za počítání s ciframi u nás arabskými zvanými. Termín *algorithmus* je vzat přímo ze jména Clho.

5.4 Celý svět modeluje

O velkých osobnostech se dočteme:

Immanuel Kant (1724 Královec - 1804 Královec) byl jeden z nejvýznamnějších evropských myslitelů. Přednášel logiku, matematiku, dynamiku, morální filosofii a metafyziku a jako úvod ke studiu filosofie fyziku, geografii a antropologii. Mimo jiné řekl:

Tvrdím však, že v každé jednotlivé nauce o přírodě lze najít jen tolik skutečné vědy, kolik je v ní matematiky.

Isaac Newton (1642 -1727) byl fyzik, matematik, astronom, filosof, teolog a alchymista. Je považován za jednoho z nejvýznamnějších vědců všech dob či dokonce za jakéhosi zakladatele moderní fyziky a vědy obecně.

Archimédés ze Syrakus (287 př.n.l. - 212 př.n.l.), byl řecký matematik, fyzik, filozof, inženýr, vynálezce a astronom. Je považován za jednoho z nejvýznamnějších vědců klasického starověku, za největšího matematika své epochy a jednoho z největších matematiků vůbec. Na poli fyziky patří mezi jeho nejslavnější objevy ve statice (mechanická rovnováha, vysvětlení principu páky) a hydrostatice (Archimédův zákon). Navrhl a sestrojil mnoho vynálezů, sloužících pro potřeby jeho rodného města Syrakus, včetně šnekového čerpadla, kterým byla vybavena největší loď starověku Syrakúsia. Mimo jiné řekl:

Dejte mi opěrný bod a já pohnu Zemí.

Klaudios Ptolemaios (asi 85 - asi 165 n.l, Alexandrie) byl řecký geograf, matematik, astronom a astrolog, který žil a pracoval v egyptské Alexandrii. Proto je znám i jako *Ptolemaios Alexandrijský*. Jeho spisy, které představovaly shrnutí starověké astronomie, astrologie a geografie, platily jako autoritativní zdroj až do novověku.

Johannes Kepler (1571 - 1630) byl německý (rakouský) matematik, astrolog a astronom. Především ve starší české literatuře se používá i počeštěná forma jeho křestního jména (Jan Kepler). Několik let působil v Praze na dvoře císaře Rudolfa II. V Praze také formuloval dva ze tří Keplerových zákonů.

Ve svých pracích se zabýval astronomií, matematikou, mechanikou, krystalografií a astrologií. V roce 1615 vyšla jeho práce *Nova Stereometria Doliorum Vinariorum*, ve které počítal objemy těles, které vznikly rotací kuželoseček kolem osy ležící v jejich rovině. Přitom použil *infinitesimálních metod* a toto dílo znamenalo významný krok ke vzniku moderních integračních metod.

Leonhard Paul Euler (1707 - 1783) byl švýcarský matematik a fyzik. Je po něm pojmenován měsíční kráter Euler. Euler je považován za nejlepšího matematika 18. století a za jednoho z nejlepších matematiků vůbec. Byl také velmi plodným autorem knih, jeho sebrané spisy čítají 60—80 svazků. Eulerův vliv na matematiku vyjadřuje výrok připisovaný Pierru Simonu de Laplaceovi:

Čtěte Eulera, čtěte Eulera, je to učitel nás všech.

Euler provedl mnoho objevů na poli diferenciálního počtu a teorie grafů. Zavedl také spoustu nových moderních matematických pojmů a symbolů, obzvláště v matematické analýze. Je také proslulý svými pracemi v mechanice, optice a astronomii.

Poté, co 17. května 1727 přijel Euler do Petrohradu, byl povýšen ze svého nižšího postu na lékařské katedře na akademii do vyšší pozice na katedře matematiky, kde úzce spolupracoval s *Danielem Bernoullim*. Euler se rychle naučil rusky a zapadl

do života v Petrohradě. Občas si také přivydělával jako zdravotník u ruského námořnictva.

Universita v Petrohradě byla založena Petrem Velikým, aby pozvedla vzdělanost v Rusku a přiblížila ho zemím západní Evropy. Akademie byla štědře financována a měla k dispozici obsáhlou knihovnu. Protože bylo na akademii zapsáno velmi malé množství studentů, výdaje na vyučování byly nízké. Akademie naproti tomu investovala do výzkumu a poskytovala jak čas, tak svobodu pro bádání o vědeckých otázkách, což přitahovalo zahraniční vědce jako byl Euler.

Znepokojen pokračujícími nepokoji v Rusku, opustil Euler 19. června 1741 Petrohrad. Dostal místo na berlínské akademii, které mu bylo nabídnuto Fridrichem II. Velikým. V Berlíně strávil 25 let a napsal zde přes 380 prací. Vydal zde dvě práce, které ho nejvíce proslavily:

Introductio in analysin infinitorum,

vydáno v roce 1748, a:

Institutiones calculi differentialis (1755)

o diferenciálním počtu.

Euler byl požádán, aby vyučoval princeznu z Anhalt-Dessau, Fridrichovu neteř. Euler jí napsal přes 200 dopisů, které byly později vydány pod názvem Eulerovy dopisy na různá témata přirozené filosofie psané pro německou princeznu. Dopisy obsahovaly Eulerovo vysvětlení různých objektů ve fyzice a matematice. Nabízejí také cenný náhled na Eulerovu osobnost a jeho náboženské přesvědčení. Tato kniha byla čtena víc než jakákoli jiná jeho matematická práce a byla vydávána v celé Evropě i ve Spojených státech. Popularita Dopisů dosvědčuje Eulerovu schopnost efektivně sdělit vědecké poznatky laické veřejnosti.

Nakonec však byl Euler navzdory svým velkým zásluhám o prestiž akademie nucen z Berlína odejít. Částečným důvodem byl spor s Fridrichem, který smýšlel o Eulerovi jako o obyčejném člověku, obzvláště v porovnání s okruhem francouzských filozofů, jako byl například *Voltaire*, které král na akademii přivedl. Euler, zbožný a pracovitý člověk, byl v mnoha ohledech přímým opakem *Voltairea*. Euler nebyl dobrý řečník, ale pouštěl se do diskuzí o problémech, o kterých mnoho nevěděl, a tak se stával častým cílem *Voltaireových* vtipů. Fridrich také vyjádřil zklamání nad Eulerovými praktickými technickými schopnostmi:

„Chtěl jsem mít na zahradě vodotrysk: Euler vypočítal sílu kol nezbytnou k čerpání vody z nádrže, odkud měla být odváděna kanály a konečně tryskat v Sanssouci. Kolo bylo zkonstruováno s geometrickou přesností a nedokázalo pracovat na vzdálenost větší než padesát kroků od rezervoáru. Nicotná ješitnost! Nicotná geometrie!“

Albert Einstein se narodil roku 1879 v Ulmu v německém Württembersku, asi 100 km východně od Stuttgartu v židovské rodině. Jeho rodiči byli Hermann Einstein (1847-1902), obchodník, který později pracoval jako elektrochemik, a jeho žena Pauline (1858-1920), rozená Kochová. Albert navštěvoval katolickou obecnou

školu a na naléhání své matky bral hodiny houslí. Když bylo Albertovi pět let, jeho otec mu ukázal kapesní kompas a Einstein poznal, že něco v „prázdném“ prostoru musí působit na střelku. Později tuto zkušenost popsal jako jednu z nejdůležitějších ve svém životě. Stavěl pro zábavu fyzikální modely a mechanická zařízení. V roce 1894, poté, co zkrachovala Hermannova elektrotechnická firma, se Einstein přestěhoval s rodiči z Mnichova do Pavie v Itálii. Albert kvůli tomu nedokončil střední školu.

Petr Vopěnka (1935 – 2015) byl především pokračovatelem výše uvedených osobností ve smyslu symbiozy matematiky, logiky a filosofie. Při pohledu na historii matematiky razil termíny „matematika manipulací a kalkulací“. Když si uvědomíme, jak velké osobnosti objevovaly přírodní zákony a jak vůbec vznikají nové technologické objevy, zjistíme, že na začátku je vždy nějaké experimentování (modelování skutečnosti - představa o problému, manipulace, kalkulace), pak ověřování a analýza. Až na konci řetězce je zobecňování a vytváření teorií a obecných matematických struktur.

5.4.1 Doby stárnou rychleji než lidé

Jak tomu rozumět? Naši předkové se narodili do doby, ve které také zemřeli. Během jejich života žádné velké změny nebyly. My už zemřeme v době, která se dost podstatně liší od doby, do které jsme se narodili (počítače, mobily, internet, atd.).

Současná mládež vůbec neví (ani se to nedá odhadnout), jaká bude doba, ve které zestárnou a umřou. Budou té době vůbec rozumět? Jmenujme aspoň některé:

- fyzika (částicová fyzika a jaderná energetika),
- chemie (nové materiály, potravinářství, zemědělství),
- biologie (zdravotnictví, genetika),
- medicína (CT, magnetická rezonance, umělé tkáňové náhrady).

Například Nobelova cena za medicínu (asi v roce 1978), byla udělena tvůrcům CT (počítačová tomografie), přitom základ byl znám již z třicátých let jako geometrický poznatek rekonstrukce obrazu a *Radonovy transformace*.

Rozvoj počítačových technologií dovilil završit celý proces. U magnetické rezonance to byl „vedlejší produkt“ rozvoje neutronové fyziky a matematické teorie neutronového záření. Takto bychom mohli pokračovat téměř „do nekonečna“.

5.4.2 Matematické modelování a vzdělávání modelářů

V současné době probíhá přímo exploze publikací o matematickém a počítačovém modelování (MPM). Vědecké a vývojové týmy, dokonce i marketinkové skupiny, se věnují problémům MPM s velkou intenzitou a ve velkém množství. Je možné najít mnohas stránkové traktáty o tom, co je MPM.

Dnes se to na univerzitách celého světa hemží novými obory zaměřenými na matematické modelování (matematické metody v „čemkoliv“ - je možné doplnit libovolný název vzdělávací nebo technologické disciplíny).

V náborových letácích univerzit, se dočteme:

Studenti si zejména prohloubí své znalosti matematiky a teoretických základů informatiky a seznámí se s jejich pokročilými aplikacemi ve vybraných oblastech technologií, verifikace a testování korektnosti různých kybernetických systémů, v oblastech vysoce náročných vědeckotechnických výpočtů, oblast modelování, simulace a optimalizace.

Absolvent je kvalifikačně připraven na výzkum, vývoj aplikace nejrůznějších pokročilých technologií vyžadujících hlubší pochopení matematických základů, včetně technologií vysoce nekonvenčních.

Získané znalosti umožňují absolventovi vysokou flexibilitu a snazší osvojení nových poznatků a technologií.

Zde si všimneme slov „je kvalifikačně připraven“, „umožňují snazší osvojení...“. Přeloženo:

- bude mít papír, bude muset ještě dál hodně studovat.

Nedomníváme se, že MPM lze vyučovat jako předmět na jakékoliv škole, avšak každá ukázka, o kterou se také snažíme, dá nějakou hlubší informaci o tom, co je MPM. Umíme si však stanovit, jaké prostředky, nástroje, dovednosti, vědomosti musí úspěšný modelář mít, neboť MPM je *multioborová kvalifikace*. Kromě toho musí mít velkou schopnost tvůrčího myšlení a týmové práce.

Autor těchto řádků byl jako mladý asistent členem týmu, který měl za úkol zjistit, jak vypadá teplotní pole v nádobě jaderného reaktoru, kterou měla Škoda Plzeň vyrábět. Bylo třeba hodně vstupních informací o co vlastně jde a co si od toho zadavatel slibuje. Bylo třeba konzultovat fyzikální a technické parametry úlohy, pak si doplnit chabé školské vědomosti o úlohách teplotních polí, poté se naučit prověřovat výsledky získané z numerických modelů. A to v době, kdy se komunikovalo s počítačem pomocí strojového kódu.

Tým se dopracoval k výsledkům asi za 3 roky. Dnes je možné takovou úlohu považovat za „školskou“ a řešit ji v rámci semestrální práce v tématu *Numerické modelování teplotního pole*.

Skoro o deset let později „omlazený“ tým měl za úkol vytvořit matematický model proudění přehřáté páry mezi lopatkami parní turbíny s cílem pomoci k vývoji parní turbíny velkého výkonu, opět pro Škoda Plzeň. Vědělo se, že výkon také ovlivňuje poloha tzv. *sonické čáry* (tj. rozhraní mezi podzvukovým a nadzvukovým prouděním páry mezi lopatkami turbíny).

Seznámili jsme se s tzv. Thomsonovou metodou, která modelovala prostor mezi lopatkami pomocí dvou částí kružnic s rozdílnými poloměry. Metoda však nedávala akceptovatelné výsledky. Také proto, že pohybový zákon páry byl modelován čistě hyperbolickým systémem matematických úloh. Testovací úlohy vedly na velké systémy lineárních rovnic (řádově 10^5 rovnic). Výzkum se pak rozštěpil na řešení úloh s neznámou vnitřní hranicí a na rychlé (paralelní) algoritmy založené na metodě více sítí (multigrid).

Těmito vzpomínkami chceme ukázat, jak rozmanitá je práce při vytváření matematických, numerických a počítačových modelů a jak je obtížné se na tuto práci připravit.

Existují snad tisíce často velmi pěkných knih, učebnic, textů, v nichž zájemce získá potřebné informace o matematickém modelování. Můžeme doporučit text:

Úvod do matematického modelování s využitím Maple, autorů Hřebíček, Pospíšil, Urbánek, MU IBA Brno 2010.

Zabývá se populačními modely, ale úvod je dostatečně obecný.

Také je nutné zmínit článek Tomáše Roubíčka *Zárné zítřky matematického modelování? Vesmír 91, 484, 2012/9.*

Velmi silně koreluje i s našimi názory.

Dovolíme si ukradnout několik pasáží, protože autor vyjádřil mnohem lépe to, co také cítíme:

... soustředím se, jak nadpis předjímá, na modelování pomocí matematiky. Takové modelování napodobuje určitý výsek reality tím, že se snaží ujasnit základní zákonitosti, např. fyzikální, mechanické, biologické, ekonomické, které jej řídí, a formulovat je matematickými vztahy.

Ty dále analyzuje, co se týká třeba existence a vlastností řešení formulovaných úloh, a navrhuje teoreticky podložené metody jejich (alespoň přibližného) řešení. Takto připravené modely pak počítačově implementuje, provádí simulace, identifikuje parametry modelu či vstupní data. Výsledky pak vizualizuje a interpretuje, případně užívá k optimalizaci...

Stalo se moderním zaklínat se matematickými modely. V inženýrství, ekonomii, fyzice, vědě o materiálech, biologii, lékařství, finančním „průmyslu“ a leckde jinde.

Přesně vzato, ve skutečnosti ale většinou jde jen o počítačové modely bez hlubších matematických základů. Zpravidla se omezují jen na jakási (mnohdy spíše předstíraná) přibližná řešení specifických rovnic, aniž je známo, jestli tyto rovnice vůbec nějaké řešení mají, či jestli přibližná metoda opravdu poskytuje řešení, které by v nějakém smyslu bylo blízké hledanému „přesnému“ řešení, kdyby takové přece jenom, čistě náhodou existovalo.

Řeklo by se tedy, že modelování je spíše jen hraní s počítači. To by ale bylo velmi zjednodušující. V mnoha případech (nebo možná, při troše optimismu, i docela často) se i ve shora nastíněném balastu skrývá rozumné jádro, které odráží realitu šikovně zjednodušujícím, ale přesto (či právě proto) užitečným způsobem.

Mnoha kompromisům se ani v seriózním matematickém modelování prostě vyhnout nelze. Pro inženýry, fyziky, ekonomy a mnohé další si matematika často budovala postavení nestravitelné a téměř nepoužitelné disciplíny, a krom nezbytného minima se naučili prakticky bez ní obejít.

V obecné veřejnosti znechucené ze základních a středních škol předmětem, kterému se tam ne zcela přesně říká „matematika“, dokonce převládá názor, že je v matematice již vše dávno hotovo z předchozích staletí - což se jeví vědecky aktivním matematikům obzvlášť absurdní.

Stále více se uplatňuje víra (dokladovatelná výsledky), že dobře formulované matematické úlohy (tedy např. dobrá fyzika) i přes svou obtížnost nakonec při jistém štěstí a hodně úsilí vedou i k dobré matematice - když ne hned, tak alespoň za pár desítek let.

Toto je mocně podpořeno neustávajícím rozvojem numerických metod a algoritmů, a ovšem i výkonností počítačů. Na většině světových univerzit se tak ke konci 20. století zakládají katedry či oddělení s náplní matematického modelování nebo se tyto iniciativy alespoň implicitně promítají do vědeckých koncepcí již existujících matematických kateder.

Tak se matematika (nebo alespoň jisté její části) stala prakticky použitelnou vědou právě zejména projekcí svých specifických částí do matematického modelování a matematické modelování samo se tak stalo jakousi křížovatkou věd. Potkávají se různé části matematiky jak navzájem (užitá analýza, numerická matematika, algebra, geometrie, diskrétní matematika), tak zejména s jinými vědami, jako jsou fyzika, materiálová věda, biologie, geologie, chemie, a samozřejmě s aplikovanými disciplínami, jako jsou inženýrství všeho druhu či ekonomie.

5.4.3 „Antimodelářská“ výuka matematiky

Výklad matematiky a přírodních věd na školách může být antimodelářský blabla něco mám v hlavě!

5.5 Podrobnosti o slavných úlohách

5.5.1 Pověst o založení Kartága

Asi v roce 890 př.n.l. vládl ve fénickém městě Tyru (jižně od Bejrútu v dnešním Libanonu) král Muttuol. Po jeho smrti připadl trůn jeho dětem – dceři Didó a synovi Pygmalionovi. Princezna Didó se pravděpodobně provdala za svého strýce Acherbase, nesmírně bohatého muže s vlivným postavením ve státě. Dnes bychom řekli, že byl vlivným a mocným lobistou. Některé historické prameny uvádějí poněkud odlišné jméno manžela princezny Didó. Uvádí se, že jejím manželem byl Sychaeus, muž stejně mocný a bohatý, jako Acherbas. Bratr princezny Didó, princ Pygmalion, však zatoužil po majetku a postavení svého švagra a tak ho nechal zabít. V té době, příznivci prince Pygmaliona, zastoupeni ve vládě, požadovali vládu pouze jednoho panovníka. Tak začalo pronásledování princezny Didó. Traduje se, že princezna stačila na poslední chvíli uprchnout s věrnou posádkou na jedné lodi a „zachránit“ značnou část státního pokladu.

Po poměrně dlouhém a strastiplném putování se posádka utábořila na pobřeží dnešního Tuniska. Díky „zachráněné“ části státního pokladu byla Didó relativně bohatá a tak chtěla od místního vládce koupit pozemek k založení osady. Místní vládce, král Hiarbo, oslněn krásou princezny Didó s prodejem pozemku na mořském pobřeží souhlasil. Vymínil si však, že „ne větší, než jaký lze ohraničit volskou kůží“. Zřejmě ale nepočítal s tím, že krásná princezna Didó může být také chytrá. Rozřezala volskou kůži na tenké proužky, které svázala a tak získala velmi dlouhý provaz dané délky. Tím pak ohraničila pozemek s maximální výměrou (plochou). Tak přelstila „chytrého“ krále Hiarba.

Na takto získaném pozemku pak založila pevnost Byrsu, která se později stala základem mocného města, dnes bychom řekli supervelmoci tehdejšího světa – Kartága.

Ať je legenda o založení Kartága pravdivá či nikoliv, historickým faktem zůstává, že bylo založeno přibližně v roce 814 př.n.l. féniciány jako kolonie Tyru. Podle pověsti byla Didó první královnou Kartága. Některé prameny uvádí jiná jména první královny, „Dídó“, „Elissar“, či „Alissa“. Ať se jmenovala jakkoliv, stala se mezi svým lidem velmi oblíbenou a Kartágo pod její vládou prosperovalo.

Jméno Kartágo je odvozené z fénického *QRT HDŠT*, neboli *Qart Hadašt* znamenající *Nové město*. Starověcí féniciané pravděpodobně nebyli příliš „tvořiví“ při pojmenovávání svých nových kolonií, jméno *Qart Hadašt* neslo více jejich zámořských osad. Proslulosti ve starověkém světě však dosáhlo pouze jedno jediné.

Postupem času Kartágo vyrostlo v obrovskou ekonomickou mocnost ve Středomoří. Shromáždilo obrovské bohatství a získalo dominantní ekonomický vliv. Ve své době vládlo dalším 300 městům kolem západní části Středozemního moře.

Zatímco termín „kartaginský“ je spíše používán současnými autory, starověké spisy ve vztahu k čemukoliv kartaginskému používají termín, „punský“. Latinský termín *punicus* odvozený od staršího *poenicus* původně pochází z řeckého „foiniké“

φοινικη ,, znamenající „Fénicie“.

Kartágo bylo ve 2. a 3. stol. př.n.l., řečeno dnešní terminologií, úřadující supervelmocí. Velmi vážného konkurenta ale měla v Římské republice, coby „nastupující“ supervelmoci. S ní soupeřila o nadvládu nad západním Středomořím. Tato rivalita posléze vyústila v sérii válek, které vešly do dějin pod označením *punské války*.

Série tří válek měla pro Kartágo zničující dopad. Bohaté Kartágo dokázalo vybudovat mocné armády se schopnými vojevůdci. Hannibal přešel se svými slony Alpy a po počátečních drtivých vítězstvích nad římskými legiemi stanul před branami Říma. Řečeno dnešní terminologií, však doplatil na hrubé podcenění logistiky. Ve chvíli, kdy stál před branami Říma, měl kritický nedostatek skoro všeho vojenského materiálu. Zásobami potravin počínaje a unavenou a vyčerpanou armádou konče. Takže nebyl schopen dobýt žádné větší opevněné římské město. Této šance Římané bezezbytku využili.

Podobné „drobnosti“ nakonec vedly k porážkám Kartaginských armád. Prohry vedly k úpadku ekonomické a politické síly Kartága. Úpadek ještě umocnily velmi tvrdé sankce uvalené na Kartágo vítězem, jako podmínka k zastavení bojů. Zda došlo k úpadku Kartága díky prohraným válkám, nebo zda úpadek města, jeho rozmařilost, způsobila, že díky tomu Kartágo prohrálo války dnes pravděpodobně nezjistíme. Třetí a poslední punská válka skončila úplným zničením města Kartága a anektování kartaginského území Římany.

Studium historie Kartága je poněkud problematické, Římané po zničení původního města vystavěli na jeho rozvalinách Kartágo nové, nyní již ale římské. Téměř všechny kartaginské památky, buď písemné, či stavební byly po vyhraných válkách Římany metodicky ničeny. Není se čemu divit, Řím většího a nebezpečnějšího nepřítele do té doby i od té doby neměl. Většina z dochovaných záznamů, či překladů původních textů, „pochází“ od římských historiků. Ti svého „úhlavního nepřítele“ neměli nejmenší důvod vychvalovat či obhajovat.

Archeologické výzkumy neustále pokračují na původních kartaginských místech a mnohé z vykopávek některé aspekty tradičního obrazu Kartága potvrzují, mnohé z nich naopak vyvracejí.

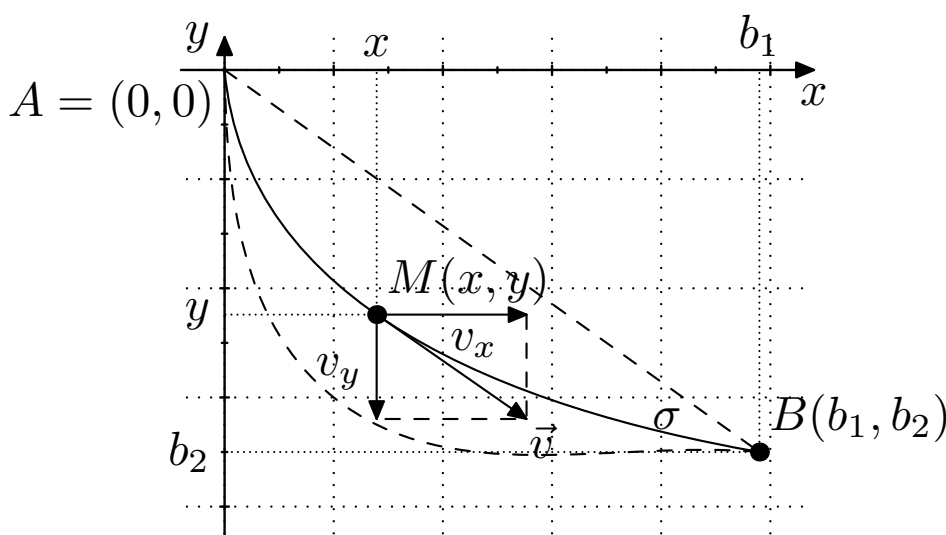
5.5.2 Brachystochrona nebo brachistochrona

Jedna ze „slavných“ úloh je *úloha o brachistochroně*. Na první pohled vypadá velmi tajemně, ale ve skutečnosti je to úloha stará více než 350 let.

V roce 1696 v pojednání Johanna Bernoulliho „*Problema novum, ad cujus solutionem mathematici invitantur*“ („Nová úloha, k jejímuž řešení zveme matematiky“) byla formulována úloha: „Ve vertikální rovině jsou dány dva body A a B . Najděte dráhu, po níž se těleso působením vlastní tíže dostane z bodu A do bodu B za nejkratší dobu.“ Tuto úlohu řešili dále Leibnitz, Jacob Bernoulli, Newton a řada dalších a stala se základem *variačního počtu*. Je pozoruhodné, že v 17.století jistě nikdo neměl ani ponětí o lyžařských můstcích nebo o vodních skluzavkách.

Úkolem je najít tvar spojnice místa A a B , po které by se těleso pohybující se vlivem gravitační síly, dostalo z místa A do B v nejkratším čase. Předpokládá se pohyb v homogenním gravitačním poli a odporové síly (tření a odpor vzduchu) se zanedbávají.

Pro základní představu si zvolíme souřadnicový systém podle obrázku 5.5.1



Obrázek 5.5.1: Možné křivky nejrychlejšího spádu

Hmotný bod o hmotnosti m se má pohybovat bez tření vlastní tíží po oblouku AMB . Rychlost $\vec{v}$ hmotného bodu místě M nezávisí na tvaru křivky spojující body A a B (Galileův zákon) a podle zákona energetické bilance je kinetická a potenciální energie hmotného bodu v každém bodě $M = (x, y)$ v rovnováze.

Předpokládáme, že každá spojnice σ bodů A a B , která přichází v úvahu je hladká křivka (nepřipouštíme např. „schody“, nebo siluetu chrámu sv. Bartoloměje v Plzni).

Poloha pohybujícího se bodu M po křivce σ se s časem mění, tj. $x = x(t)$ a $y = y(t)$. Čas potřebný k dosažení bodu B závisí na křivce σ , označíme jej $T(\sigma)$. Označíme ještě $\mathcal{S}$ množinu všech přípustných křivek σ , tj. takových, které splňují koncové podmínky (viz obrázek 5.5.1):

$$x(0) = 0, \quad x(T_B) = b_1; \quad y(0) = 0, \quad y(T_B) = b_2.$$

Formulace úlohy:

Chceme stanovit takovou křivku $\sigma \in \mathcal{S}$, pro níž je $T(\sigma) = \min T(\sigma) \forall \sigma \in \mathcal{S}$.

Je patrné, že klíčovým problémem je určení závislosti času na křivce σ , tj. funkce $T(\sigma)$. K tomu je ovšem zapotřebí náročnější poznatky z diferenciálního a integrálního počtu.

Konkrétnímu výpočtu budeme věnovat pozornost v sešitu věnovaném optimalizačním technologiím.

Podobně jako v předchozím odstavci i zde optimalizujeme funkci, jejíž argumentem je množina (křivka).

Poznámka:

Ke sporu o gramaticky správné znění úlohy uvedené, jako titulek odstavce, lze v různých zdrojích „na Internetu“ nalézt²³ příslušná vysvětlení. Z nich plyne, že:

Brachistochrona (z řeckého brachistos - nejkratší, chronos - čas), označovaná také jako křivka nejkratšího spádu, je křivka spojující dva body, po které se hmotný bod dostane z počátečního klidu v jednom bodě do druhého působením homogenního gravitačního pole za nejkratší dobu.

Některé anglosaské oblasti Evropy⁴ však připouští obojaký zápis jako gramaticky správný. Tedy jak *brachystochrona*, tak i *brachistochrona*.

²<https://cs.wikipedia.org/wiki/Brachistochrona>

³http://dml.cz/bitstream/handle/10338.dmlcz/138956/PokrokyMFA_17-1972-4_8.pdf

⁴Ottův naučný slovník, příp. francouzsky mluvící země. . .

Seznam obrázků

1.2.1	Možné tvary oblasti ohraničené danou délkou oblouku l_D	12
1.2.2	Možné tvary oblasti ohraničené danou délkou oblouku l_D	12
1.2.3	7 Mostů v Königsbergu - historická mapa	14
1.2.4	7 Mostů v Königsbergu - idealizace	14
1.2.5	7 Mostů v Königsbergu - idealizace	14
1.2.6	7 Mostů v Königsbergu - Eulerův graf	14
2.1.1	Příklad algoritmu (stavového diagramu)	22
2.5.2	Definice časové složitosti	34
3.2.1	Řešení funkcionální úlohy $x - y^2 = 0$	42
3.2.3	Dvě ze spojitých řešení rovnice $y^2 - (x^2 - 1)^2 = 0$	44
3.2.4	Jedno nespojitě řešení rovnice $y^2 - (x^2 - 1)^2 = 0$; $k = \frac{1}{2}$	45
3.3.5	Grafické řešení soustavy rovnic	48
4.1.1	Zákon $p = d(Q_D)$	55
4.1.2	Zákon $Q_D = D(p)$	55
4.1.3	$p = s(Q_S)$	56
4.1.4	$Q = S(p)$	56
4.1.5	Rovnováha nabídky a poptávky	57
4.1.6	Pavučinový model zákona nabídky a poptávky	58
5.5.1	Možné křivky nejrychlejšího spádu	78

Seznam tématických bloků

2.1.1	Možná pojetí algoritmu	20
2.5.2	Funkce algoritmu <i>quicksort</i>	31
2.5.3	Činnost funkce <i>rozděl</i>	32
2.5.4	Definice časové složitosti	33
2.5.5	Třída časové složitosti	33
3.1.1	Lineární rovnice pro jednu neznámou	37
3.1.2	Polynomiální rovnice	38
3.1.3	Rovnice v $\mathbb{R}$	39
3.2.4	Lineární rovnice pro více neznámých	41
3.2.5	Funkcionální rovnice v $\mathbb{R}^2$	43
3.2.6	Globální a lokální řešení	43
3.3.7	Přímka	45
3.3.8	Kružnice	45
3.3.9	Elipsa	46
3.3.10	Rovina	46
3.4.11	Ilustrace metody řešení „nedourčené“ soustavy	48
3.4.12	Základní úloha lineární algebry	50
3.4.13	Řešitelnost úlohy $\mathbf{Ax} = \mathbf{b}$	52