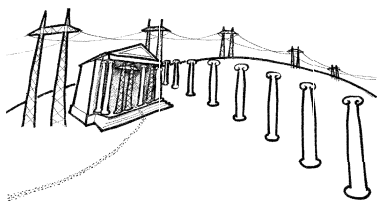


INFORMATIKA

Společná vybraná podposloupnost (Úlohy z MO – kategorie P, 17. část)

PAVEL TÖPFER

Matematicko-fyzikální fakulta UK, Praha



Tentokrát se vrátíme sedm let dozadu do historie matematické olympiády – kategorie P. V krajském kole 49. ročníku (školní rok 1999/2000) byla zadána také klasická programátorská úloha na nalezení společné vybrané podposloupnosti s maximálním souč-

tem. Její řešení představuje typickou ukázkou toho, k čemu se hodí technika dynamického programování. Jako obvykle se nejprve seznámíme se zadáním úlohy v původním znění:

Nechť $A = (a_1, a_2, \dots, a_M)$ je posloupnost celých čísel. Posloupnost A' nazveme vybranou podposloupností této posloupnosti, jestliže vznikne z posloupnosti A vynecháním některých jejích členů, přičemž pořadí ostatních prvků zachováme. Společná vybraná podposloupnost posloupností A , B je každá taková posloupnost C , která je vybranou podposloupností obou posloupností A i B .

Příklad

Nechť $A = (1, 11, 2, 1, 4, 99)$, $B = (9, 4, 1, 2, 7, 1, 99)$. Posloupnost $(1, 2, 1, 99)$ je společnou vybranou podposloupností posloupností A , B . Po-

sloupnost $(1, 2, 4)$ je vybranou podposloupností posloupnosti A , ale není vybranou podposloupností posloupnosti B , takže není ani společnou vybranou podposloupností A a B .

Soutěžní úloha

Máme dány dvě posloupnosti kladných celých čísel $A = (a_1, a_2, \dots, a_M)$ a $B = (b_1, b_2, \dots, b_N)$ s délkami M , resp. N . Tyto posloupnosti jsou uloženy v polích, jejichž obsah není dovoleno měnit. Uvažujme takovou společnou vybranou podposloupnost posloupností A a B , ve které součet všech jejích členů je největší možný. Napište program, který vypíše součet všech členů takovéto společné vybrané podposloupnosti.

Poznámka

Existuje algoritmus, který tuto úlohu řeší v čase úměrném $M \times N$ a pomocné paměti, jejíž velikost je úměrná menšímu z čísel M , N .

Příklad

Vstup: $M = 6$, $N = 7$

$A = (1, 11, 2, 1, 4, 99)$

$B = (9, 4, 1, 2, 7, 1, 99)$

Výstup: 103

Společné vybrané podposloupnosti se součtem 103 existují dokonce dvě: $(4, 99)$ a $(1, 2, 1, 99)$.

Primitivní řešení této úlohy spočívá v tom, že budeme postupně generovat všechny podposloupnosti vybrané z posloupnosti A , pro každou z nich ověříme, zda je také vybranou podposloupností posloupnosti B a pokud ano, zjistíme její součet. Z takto získaných součtů si průběžně udržujeme dosažené maximum, které je pak výsledkem úlohy. Tento postup je teoreticky naprosto správný, neboť systematicky prozkoumá všechny společné vybrané podposloupnosti obou zadaných posloupností A , B . Programová realizace rovněž není obtížná – generování všech vybraných podposloupností posloupnosti A zajistí nejlépe rekursivní procedura pracující s hloubkou rekurze M a zbývající operace jsou již triviální. Podstatnou závadou uvedeného postupu je však příliš velká časová složitost. Z posloupnosti A délky M lze totiž vybrat až 2^M různých podposloupností, neboť pro každý

prvek posloupnosti A máme vždy dvě možné volby – zda ho do právě vybrané podposloupnosti zařadíme nebo nezařadíme. Algoritmus má tedy exponenciální časovou složitost vzhledem k délce posloupnosti A . Alespoň nějaké časové optimalizace dosáhneme, když za A vezmeme kratší z obou zadaných posloupností, na exponenciální složitosti tohoto řešení to však nic nezmění.

Zapomeneme tedy raději na slepé zkoušení všech možností a k řešení naší úlohy přistoupíme jinak. Využijeme metodu dynamického programování. Označme si symbolem A_i posloupnost prvních i členů posloupnosti A , tzn. $A_i = (a_1, a_2, \dots, a_i)$. Podobně B_j bude znamenat posloupnost prvních j členů posloupnosti B , tedy $B_j = (b_1, b_2, \dots, b_j)$. Nyní si zformulujeme obecnější úlohu: pro každou dvojici čísel i, j ($0 \leq i \leq M$, $0 \leq j \leq N$) určíme maximální součet prvků společné vybrané podposloupnosti z posloupností A_i a B_j . Budeme postupně řešit tyto obecnější úlohy pro různé dvojice i, j a výsledky si budeme ukládat do dvojrozměrné tabulky $P[0..M, 0..N]$. Hodnota $P[i, j]$ bude vždy představovat řešení úlohy pro posloupnosti A_i a B_j . Jak je obvyklé u dynamického programování, při řešení budeme postupovat od úloh jednodušších k úlohám složitějším, tedy od menších hodnot i, j k hodnotám větším. Při řešení větších úloh přitom budeme využívat již známé výsledky menších úloh uložené v tabulce P . Výsledkem celé původní úlohy bude hodnota $P[M, N]$.

Tabulku P budeme vyplňovat po řádcích. Při výpočtu hodnot na i -tém řádku přitom budeme využívat již spočítané hodnoty z předchozího řádku $i - 1$. Řádek tabulky s indexem 0 bude obsahovat samé nuly, neboť z prázdné posloupnosti nelze vybrat žádnou podposloupnost. Řádek s indexem i (pro $i > 0$) vyplníme podle předchozího řádku s indexem $i - 1$ následovně. Prvek $P[i, 0]$ bude mít zřejmě hodnotu nula. Prvek $P[i, j]$ (pro $j > 0$) dokážeme spočítat na základě již známých hodnot $P[i - 1, j]$, $P[i, j - 1]$ a $P[i - 1, j - 1]$. Pokud se čísla a_i a b_j neshodují, každá společná vybraná podposloupnost posloupností A_i a B_j je zároveň buď vybranou podposloupností posloupností A_{i-1} a B_j , nebo je vybranou podposloupností posloupností A_i a B_{j-1} . V tomto případě je tedy prvek $P[i, j]$ roven maximu z čísel $P[i - 1, j]$ a $P[i, j - 1]$. Jestliže platí $a_i = b_j$, potom jednou z možných společných vybraných podposloupností posloupností A_i a B_j s maximálním součtem je společná vybraná podposloupnost posloupností A_{i-1} a B_{j-1} s maximálním součtem doplněná o přidaný poslední člen $a_i = b_j$. Hodnota $P[i, j]$ je v tomto případě tudíž rovna $P[i - 1, j - 1] + a_i$.

Popsaný algoritmus má časovou složitost $O(M \times N)$, neboť je třeba

postupně spočítat $M \times N$ prvků tabulky P a výpočet každého z nich provedeme v konstantním čase. Paměťová složitost výše popsané implementace algoritmu je rovněž $O(M \times N)$, což je velikost tabulky P . Pro stanovení výsledné hodnoty $P[M, N]$ ovšem ve skutečnosti celou tabulku P nepotřebujeme ukládat do paměti. Každý řádek tabulky P závisí pouze na bezprostředně předcházejícím řádku, takže stačí pamatovat si vždy jen její poslední dva řádky. Při šikovné programové realizaci vystačíme dokonce jen s jediným jednorozměrným polem velikosti N , ve kterém se postupně střídají hodnoty prvků z jednotlivých řádků tabulky. Takový způsob řešení ukazuje i následující programová ukázka. Paměťová složitost výsledného programu je tedy pouze $O(M + N)$. Kromě prostoru na uložení zadaných posloupností čísel A, B využívá program pomocnou paměť pouze na jeden řádek tabulky P , tedy paměť, jejíž velikost je rovna menšímu z čísel M, N (pokud by bylo $M < N$, stačí zaměnit pořadí obou zadaných posloupností).

program Podposloupnost;

```

const MAX = 1000;           {maximální délka posloupnosti}

var A, B: array[1..MAX] of integer;           {zkoumané posloupnosti}
    M, N: integer;           {délky posloupností A, B}
    P: array[0..MAX] of integer;           {řádek tabulky}
    Pred: integer;           {předchozí hodnota P[i-1,j-1] při
                                výpočtu P[i,j]}
    X: integer;           {uschování hodnoty pro Pred}
    i: integer;           {počítaný řádek tabulky P}
    j: integer;           {počítaný prvek v řádku P}

begin
write('Počet prvků posloupnosti A: ');
readln(M);
write('Prvky posloupnosti A: ');
for i:=1 to M do read(A[i]);
writeln;
write('Počet prvků posloupnosti B: ');
readln(N);

```

```

write('Prvky posloupnosti B: ');
for i:=1 to N do read(B[i]);
writeln;

for j:=0 to N do P[j]:=0;    {řádek tabulky P číslo 0}
for i:=1 to M do            {i-tý řádek tabulky P}
  begin
    Pred:=0;
    for j:=1 to N do {počítáme P[j] z i-tého řádku}
      begin
        X:=P[j];          {uschování dosavadního P[j]}
        if a[i] = b[j] then
          P[j]:=Pred+a[i]    {nová hodnota P[j]}
        else
          if P[j-1]>P[j] then P[j]:=P[j-1];
          Pred:=X             {hodnota Pred pro další prvek}
        end
      end
    end;

writeln('Maximální součet společné vybrané
        podposloupnosti: ', P[N])

end.

```

Nejnovější verze Cabri 3D

O prázdninách se na www.cabri.com (jen ve francouzské větvi) objevila ještě novější verze 2.1 programu Cabri 3D, než o které jsme referovali v minulém čísle MFI (roč. 17, 2007, č. 1, s. 52). Přináší další významnou inovaci: možnost vytvoření bodu zadáním tří souřadnic, měření povrchu válce a kužele (nejen rotačních), skalární a vektorový součin, stejnolehlost, kulovou inverzi, protokol o konstrukci. Program zatím komunikuje francouzsky a anglicky, česká lokalizace i česká příručka budou brzy k dispozici. Varování pro zájemce, kteří si budou novou verzi instalovat: Starou verzi je třeba nejprve schovat nebo přejmenovat, jinak ji instalační program zlikviduje.

Antonín Vrba

Graf funkce s parametry v Excelu

VÁCLAV MATYÁŠ

Purkyňovo gymnázium, Strážnice

Úvod

Excel dovede rychle a bezchybně vykreslovat grafy funkcí, o kterých se studenti učí na střední škole. Navíc umožňuje studentům (v rámci konstruktivistického přístupu) samostatně vyšetřovat vzhled grafů v závislosti na měnících se parametrech ve funkčních předpisech. Ve srovnání s dobou, kdy byl psán článek [1], jsou nyní možnosti Excelu mnohem výraznější. Změny parametrů lze řešit jednoduše pomocí *posuvníků*; ukážeme si poměrně podrobně základní postup jejich zavedení a použití při zobrazování grafů.

Lineární funkce $y = kx + q$

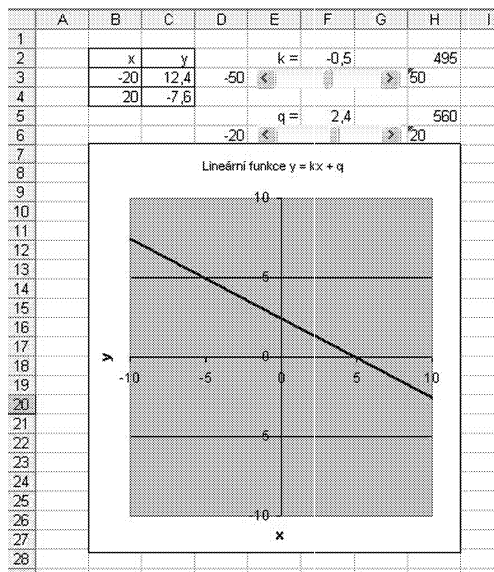
Navrhujeme zde postup, který vede k obr. 1. Uživatel si může ovšem podle svých potřeb volit jiné rozsahy proměnných i jiné uspořádání prováděných výpočtů, ale o to nyní nejde.

Excel vykresluje graf funkce na základě tabulky funkčních hodnot, takže je třeba začít tabulkou. Z praktických důvodů je vhodné změnit šířku sloupců A až H na údaj 6 (volbou *Formát – Sloupec – Šířka*). Do buněk B2 a C2 zapíšeme jako text záhlaví tabulky: x a y . V případě lineární funkce víme, že grafem je přímka, k jejímuž sestrojení stačí dva body, takže zvolíme dvě různé hodnoty nezávisle proměnné x , do B3 např. 20, do B4 vložíme 20, což budou hodnoty mimo rozsah budoucího obrázku. Hodnoty y do sloupce C tabulky vložíme, až si je připravíme.

Parametr k (nebudeme vyjímát hodnotu 0 pro funkci konstantní) je zadáván pohybem ukazatele na posuvníku [3]. Obrázek posuvníku vytvoříme klepnutím pravým tlačítkem myši na volbu *Formát* v roletové nabídce Excelu, pak levým tlačítkem zadáme volbu *Formuláře*, ve svislé tabulce klepneme levým tlačítkem na ikonu posuvníku (tento popis se nám po nastavení kurzoru na ikonu vypíše) a pak znovu levým na plochu Excelu. Tím se posuvník (ve svislé poloze) a s vyznačením (kolečka na okraji obrázku) objeví na ploše. V ploše posuvníku se kurzor změní na kříž a po stisknutí levého tlačítka myši můžeme celým posuvníkem v ploše posuno-

vat. Po uchopení levým tlačítkem za označení posuvníku (kříž kurzoru se změni na dvojšipku) můžeme pohybem kurzoru změnit polohu (svislá – vodorovná) a velikost posuvníku. Označení posuvníku můžeme kdykoliv zrušit stisknutím levého tlačítka myši mimo obrázek posuvníku a naopak můžeme posuvník kdykoliv označit stisknutím pravého tlačítka v obrázku posuvníku. (Označený posuvník můžeme zrušit zadáním volby *Vyjmout* myši nebo *Delete* z klávesnice.)

Pro náš obr. 1 umístíme posuvník do 3. řádku přes buňky E3, F3, G3. Máme-li obrázek posuvníku již nastaven v požadované velikosti a poloze, zadáme krajní hodnoty parametru k (zde např. od -50 do 50) a název buňky, ve které se bude zobrazovat aktuální mezihodnota pro parametr k nastavená na posuvníku (zde H2). Označíme posuvník (není-li), stiskneme pravé tlačítko a volíme *Formát ovládacího prvku*. Objeví se tabulka, v níž zadáme hodnoty 0 pro volbu *Nejnižší hodnota*, 1000 pro volbu *Nejvyšší hodnota*, 1 pro volbu *Přírůstková změna* a označení buňky H2 pro řádek *Propojení s buňkou*. Při pohybu posuvníkem se tedy bude v H2 objevovat mezihodnota pro parametr k v intervalu 0; 1000 a s krokem 1.



Obr. 1

Excel nepovolí zadání záporných hodnot krajních bodů ani přírůstkovou změnu v číslech s desetinnou čárkou. My však chceme, aby se parametr k měnil v intervalu $\langle -50; 50 \rangle$ a s citlivostí na desetinu jednotky a proto musíme posuvníkovou mezihodnotu přepočítat. Teprve do buňky F2 proto budeme vypočítávat aktuální hodnoty parametru k podle vzorce pro lineární transformaci $=0,1*H2-50$, která je dána body $[0; -50]$ a $[1000; 50]$ (tj. podle přímky $y = ax + b$, kde směrnice $a = [50 - (-50)] / (1000 - 0) = 0,1$; $b = 50 - 0,1 \cdot 1000 = -50$). Pro větší přehlednost ještě zapíšeme do buňky D3 krajní hodnotu -50 , do H3 jako text krajní hodnotu 50 a do E2 text $k=$.

Tentýž postup volíme pro parametr q . Pro obr. 1 umístíme posuvník do 6. řádku přes buňky E6, F6, G6. Máme-li již obrázek posuvníku v požadované velikosti a poloze, zadáme krajní hodnoty pro parametr q (zde např. od -20 do 20) a název buňky, ve které se bude zobrazovat aktuální mezihodnota pro parametr q nastavená na posuvníku (zde H5). Do buňky F5 budeme vypočítávat aktuální hodnoty parametru q podle vzorce pro lineární transformaci $=0,04*H5-20$, která je dána body $[0; -20]$ a $[1000; 20]$ (tj. podle přímky $y = ax + b$, kde směrnice $a = [20 - (-20)] / (1000 - 0) = 0,04$; $b = 20 - 0,04 \cdot 1000 = -20$). Opět pro větší přehlednost ještě zapíšeme do buňky D6 krajní hodnotu -20 , do H6 jako text krajní hodnotu 20 a do E5 text $q=$.

Buňka F2 tak obsahuje aktuální hodnotu parametru k , kterou můžeme v požadovaném rozsahu a s požadovaným krokem měnit pohybem posuvníku. (Jemnějšího, ale pomalejšího posunu na posuvníku dosáhneme poklepáváním na šipky na jeho okrajích.) Podobně buňka F5 obsahuje q .

Nyní můžeme doplnit tabulku. Do buňky C3 zadáme vzorec $=F2*B3+F5$ a stejně tak do C4 zadáme $=F2*B4+F5$, čímž vypočítáme do C3, C4 y -ové hodnoty k daným hodnotám x . Tím jsou y -ové souřadnice obou bodů zobrazované přímky svázány s hodnotami parametrů k , q nastavovanými svými posuvníky.

Tabulku zarámujeme stejně, jak to dělá Word, tj. vyznačíme rámovanou oblast a kliknutím levou myší zadáme volby *Formát, Buňky, Ochraničení* a zvolíme typ orámování (Excel ukazuje možnosti). Chceme-li volit počet desetinných míst pro čísla v tabulce, vyznačíme oblast, v níž chceme formát čísel zadat, levým tlačítkem myši zadáme volby *Formát, Buňky, Číslo, Druh*: číslo, zápisem nebo šipkami nastavíme požadovaný počet desetinných míst a potvrdíme OK.

A nyní vytvoření obrázku. Vyznačíme oblast tabulky, z níž chceme obrázek kreslit (v našem případě celou tabulku 2×2 i se záhlavím), levým tlačítkem myši klikneme na ikonu *Průvodce grafem* a zadáme: *Typ grafu*: XY bodový, vybereme *podtyp grafu* (v našem případě: *Bodový s datovými body spojenými pomocí spojnic a bez značek*) a zadáme *Další*. U tak malé tabulky, jaká je na obr. 1 Excel někdy v další tabulce *Průvodce grafem* (tj. tabulce 2/4) chybně označí jako *Oblast dat: Řádky* a v tom případě kliknutím převalíme na *Sloupce* a zadáme *Další*. V tabulce 3/4 zadáme název grafu ($y = kx + q$), označíme osy x , y a zadáme *Další* resp. *Dokončit*.

Excel nakreslí obrázek, který ještě upravíme podle svých představ. Jestliže klikneme pravým tlačítkem myši na čísla popisující osu x a zadáme volbu *Formát osy*, ve volbě na záložce *Měřítko* zadáme (pro obr. 1), *Minimum* (-10), *Maximum* (10) a krok značení – Hlavní jednotka (5 pro obr. 1) pro osu x . Podobným způsobem můžeme po kliknutí pravým tlačítkem na některé číslo popisující osu y upravovat její měřítko a popisy; pro obr. 1 zadáme totéž jako pro osu x . V téže tabulce můžeme na záložce *Písmo* volit velikost, typ a barvu popisů os a na záložce *Číslo* počet desetinných míst pro čísla popisující osy.

Klikneme-li pravým tlačítkem myši na čáru grafu (na obr. 1 přímku), můžeme ve volbě *Formát datové řady* na záložce *Vzorky* zadat styl, barvu a šířku zvolené čáry grafu (např. zvolit její větší tloušťku) a zrušit (bylo-li kreslení bodů tabulky při tvorbě grafu původně zadáno) resp. naopak zvolit kreslení, velikost a formát (tvar) bodů tabulky (Excel napovídá možnosti). Podobně můžeme po kliknutí pravým tlačítkem myši do oblasti popisů os, legendy a nadpisu grafu upravovat jejich formáty a polohu.

Velikost (délku) jednotky na ose x resp. y vykreslovaného grafu můžeme ovlivnit jednak výše uvedenými příkazy ve volbě *Formát osy* (tj. zadáním minimální a maximální popisované hodnoty na ose x resp. y), jednak tím, že kliknutím pravým tlačítkem myši obrázek označíme a tažením za některý z jeho rohů (kurzor je ve tvaru dvojšipky) upravíme velikost a poměr jeho stran (podobně jako u posuvníku). Zpravidla tím chceme dosáhnout toho, aby jednotky na obou osách byly stejné a aby tedy u studentů vznikly standardní představy o grafech (např. že příмка $y = x$ je osou I. a III. kvadrantu).

Označený obrázek můžeme posunovat v ploše Excelu (kurzor ve tvaru křížku), opět podobně jako u posuvníku.

Jestliže nyní měníme pomocí posuvníků parametry k a q , mění se odpovídajícím způsobem graf, a studenti se užitím této úlohy mohou věnovat

matematice a vyšetřování vztahu mezi parametry a polohou zobrazené přímky.

Podobným způsobem, jako byl výše popsán, jsou zadávány posuvníky k vytvoření obrázků i u dalších grafů funkcí.

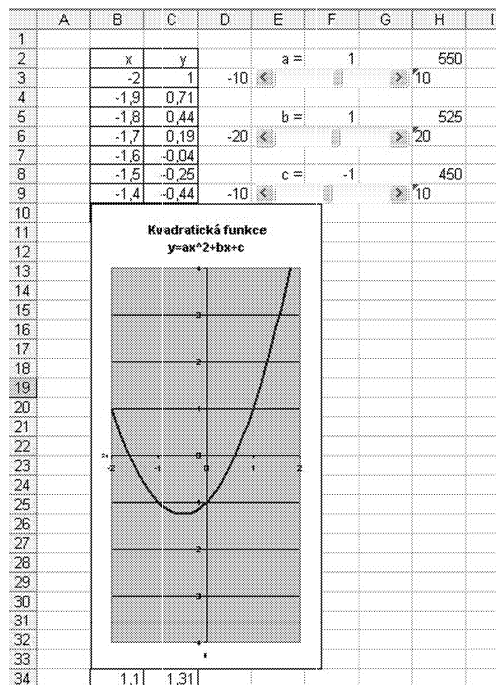
Kvadratická funkce $y = ax^2 + bx + c$

Pro lineární funkci jsme si „normální“ postup trochu zjednodušili vzhledem k tomu, že jsme jako graf funkce očekávali přímku. U jiných funkcí si však musíme pořídit obsáhlejší tabulku; lze pak zvolit takový způsob zobrazení, že Excel spojí vypočtené body a křivku „vyhladí“.

Pro vykreslení grafu kvadratické funkce (paraboly) vytvoříme tabulku, do jejíhož záhlaví opět umístíme písmeno x do B2 a y do C2. Šířku sloupců tabulky opět zúžíme na 6. Chceme-li nyní například dle obr. 2, aby nezávisle proměnná x rostla od hodnoty -2 s krokem $0,1$ po hodnotu 2 , zadáme odpovídající aritmetickou posloupnost do sloupce B tabulky obr. 2 takto: do buňky B3 zapíšeme jako číslo hodnotu -2 a do B4 zadáme vzorec $=B3+0,1$, takže se nám v B4 objeví číslo $-1,9$, stiskneme levé tlačítko myši v dolním pravém rohu buňky B4 a tažením vyplníme levý díl tabulky až po B43, kde se vypíše hodnota 2 (Vyplňujeme-li odhadem a „přetáhneme“, vymažeme sloupec přebytečných hodnot po označení klávesou *Delete*). Tabulku můžeme vytvořit i nadvakrát, nejprve po 0 (pokud se nám neukáže v buňce B23 hodnota 0 , opravíme ji na 0 , do buňky B24 zadáme vzorec $=B23+0,1$ a tento vzorec opět tažením rozšíříme až po B43).

Pak vytváříme posuvníky pro koeficienty a , b a c a umístíme je do řádků 3, 6 a 9 do buněk ve sloupcích E, F a G. V buňkách H2 resp. H5 resp. H8 nastavujeme mezihodnoty posuvníků vytvořených podobně jako v předchozí části, a v buňkách F2 resp. F5 resp. F8 hodnoty a , b , c . Volíme (dle obr. 2) $a \in \langle -10; 10 \rangle$, $b \in \langle -20; 20 \rangle$, $c \in \langle -10; 10 \rangle$, takže do uvedených buněk F vkládáme $=0,02*H2-10$, resp. $=0,04*H5/20$, resp. $=0,02*H8-10$.

Do 1. buňky tabulky C3 pak vložíme hodnotu funkce: $=\$F\$2*B3^2 + \$F\$5*B3 + \$F\8 . Ve vzorci jsou adresy buněk F zadány jako absolutní (se znaky $\$$), protože chceme, aby hodnoty odpovídajících koeficientů a , b , c zůstávaly při vyplňování sloupce C tažením neměnné, adresa B3 je naopak zadána jako relativní, protože odpovídající x -ová souřadnice bodů grafu se má při stejné akci měnit. Po zadání vzorce se do C3 vypíše hodnota dle nastavení posuvníků a tažením za pravý dolní roh této buňky vyplníme sloupec C pro všechna x zadaná ve sloupci B.



Obr. 2

Při zadávání grafu v obr. 2 byl ve volbě *Průvodce grafem*, *Typ grafu*: (XY bodový), vybrán *podtyp grafu* *Bodový s datovými body spojenými pomocí hladkých spojníc a bez značek*. (Kdybychom požadovali vykreslení bodů tabulky v grafu, volili bychom jinou příslušnou možnost.) Nakonec opět tak, jak bylo uvedeno pro lineární funkci, upravíme v obrázku rozsahy x a y a sjednotíme měřítka na osách.

Když studenti vyšetřují vztahy mezi parametry a tvarem a polohou grafu, zjistí, že v případě koeficientu b a též c pro $b \neq 0$ není tento vztah názorný ani jednoduchý a nelze ho rozpoznat pokusy na počítači. Je proto vhodné přejít ke „klasice“ a pro $a \neq 0$ si odvodit:

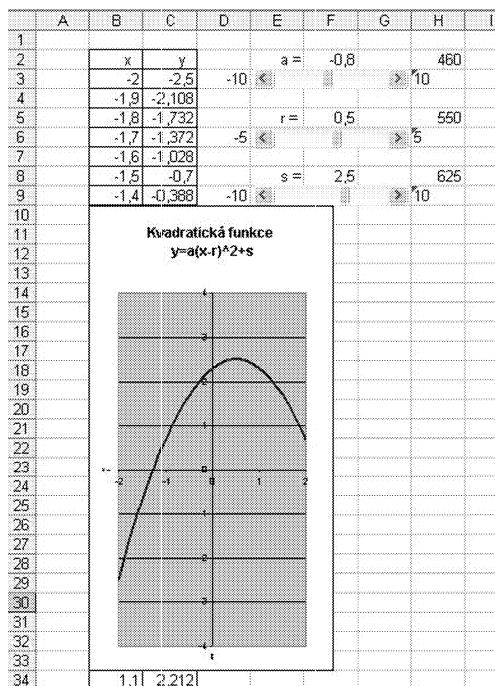
$$ax^2 + bx + c = a \left(x + \frac{b}{2a} \right)^2 + c - \frac{b^2}{4a} = a(x - r)^2 + s,$$

kde $[r; s]$ jsou, jak se ukáže, souřadnice vrcholu paraboly. Vztahy

$$r = -\frac{b}{2a}, \quad s = c - \frac{b^2}{4a}$$

skutečně nebylo možno odvodit manipulací s posuvníky. (Všimněme si také, že pro $a = 0$ dostáváme funkci lineární nebo konstantní.)

Pro studenty může být docela instruktivní zabývat se kvadratickou funkcí i ve tvaru $y = a(x - r)^2 + s$. Vytvoření jejího grafu (obr. 3) neobsahuje už žádné další problémy.



Obr. 3

Tímto způsobem můžeme znázorňovat grafy i dalších funkcí. Např. vykreslení funkce $y = d + a \cdot \sin[b(x + c)]$ s tím, že parametry a , b , c , d měníme na posuvnících a sledujeme jejich vliv na průběh kreslené

křivky např. pro $x \in \langle 0^\circ; 360^\circ \rangle$, není složitější než předchozí příklad kreslení paraboly. Tato funkce nemá body nespojitosti, její funkční hodnoty se mění v intervalu $\langle d + a; d - a \rangle$ a pro její význam např. ve fyzice je užitečné ji takto vykreslit. Je však třeba dávat pozor na to, že Excel pracuje s úhly v obloukové míře, takže chceme-li pracovat v míře stupňové, je třeba velikost úhlu (zadaného např. v B3) přepočítat na radiány a pak hodnotu sin dostaneme v C3 uložení vzorce $=\text{SIN}(\text{PI}()/180*\text{B3})$ nebo jednodušeji $=\text{SIN}(\text{RADIANS}(\text{B3}))$. Funkci SIN najdeme v roletové nabídce Excelu, zadáme-li (vše levým tlačítkem myši) *Vložit*, vybereme *Funkce* a v tabulce *Vybrat kategorii* funkce *Matematické*, pak z abecedně řazeného seznamu *Vybrat funkci* vybereme SIN(), potvrdíme OK a na řádek v nabídnuté tabulce *Číslo* (argumenty funkce) zapíšeme $\text{PI}()/180*\text{B3}$ nebo RADIANS(B3).

Při kreslení většího počtu křivek do jednoho obrázku lze s výhodou použít barevné odlišení křivek. (Excel je sám navrhne, navíc po kliknutí pravým tlačítkem myši na určitou křivku grafu můžeme ve volbě *Formát datové řady* zadat jinou požadovanou barvu čáry.)

Literatura

- [1] *Hvorecký, J.*: Tabulkové kalkulátory a grafy. MFI, r. 3 (1993/94), č. 1, s. 41 – 47.
- [2] *Brož, M. – Brožová, P.*: Excel, základní průvodce uživatele. Computer Press, Praha, 1996.
- [3] *Pecinovský, J.*: Excel 2002, podrobný průvodce pokročilého uživatele. Grada, Praha, 2002.

Z HISTORIE

Od termodynamiky Rudolfa Clausia k počátkům kvantové fyziky Maxe Plancka

K zásadnímu rozvoji termodynamiky došlo ve 2. polovině 19. století. Motivem tu bylo nalezení souvislosti mezi pohybovou energií molekul a teplem, vedoucí ke

snaze vysvětlit tepelné děje ryze mechanicky. Současně ale vyvstal problém při zjištění, že za určitých podmínek nelze teplo jako energii v soustavě znovu využít. Zákon zachování energie sice splněn byl, ale přeměna části tepla na jinou formu energie byla v určitých případech vyloučená. Jak se ukazovalo, mezi tepelným pohybem molekul a pohybem makroskopických těles přece jen odlišnosti jsou.

Základními zákony termodynamiky jsou tři věty. První je aplikací zákona zachování energie na tepelné jevy a v podobě