

## **Obsah**

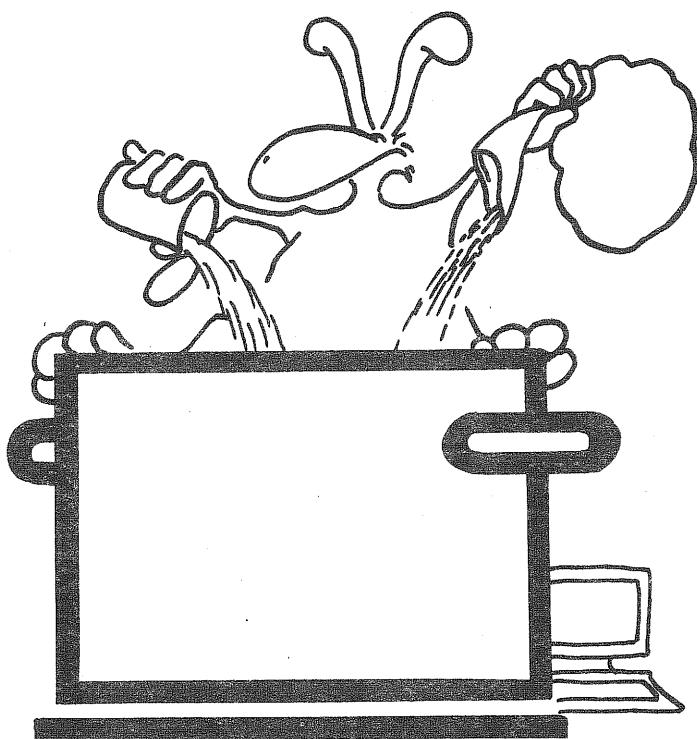
<b>9. NÁVRH PROGRAMU .....</b>	<b>315</b>
9.1. Uspořádání grafických programů .....	316
9.1.1 Fáze vývoje programů .....	316
9.1.2 Omezení efektivity .....	320
9.2. Interaktivní metody .....	322
9.2.1 Výběr podle menu .....	322
9.2.2 Světelná pera .....	326
9.2.3 Grafické digitalizační podložky (tablet) .....	328
9.2.4 Analogové pákové ovladače - joysticky .....	332
9.2.5 Samostatné elektronické kursory - myši .....	332
9.3. Konstruování obrázku .....	333
<b>10. PODNIKOVÁ GRAFIKA .....</b>	<b>343</b>
10.1. Obecné techniky .....	344
10.2. Srovnávací grafy .....	357
10.3. Vícenásobné formáty .....	372
10.4. Grafy řízení projektů .....	376
<b>11. VZDĚLÁVACÍ GRAFIKA .....</b>	<b>381</b>
11.1. Výcvikové a cvičící programy .....	382
11.2. Konzultační a dotazovací programy .....	387
11.3. Simulační programy .....	388
11.4. Počítačem řízené vyučování .....	391
<b>12. OSOBNÍ GRAFIKA .....</b>	<b>393</b>
12.1. Grafika v domácím hospodaření .....	394
12.2. Hraní her .....	406

## APLIKACE

(Jak můžeme používat grafiku ?)

V minulých částech jsme zahájili přehledem aplikačních oblastí grafiky. Nyní, po prozkoumání různých metod, které nám poskytují nástroje k vytváření grafických zobrazení, se můžeme podívat na aplikace detailněji. Nejprve budeme uvažovat efektivní metody navrhování grafických programů, pak budeme diskutovat specifické aplikace v podniku, škole či v domácnosti.

## 9. NÁVRH PROGRAMU



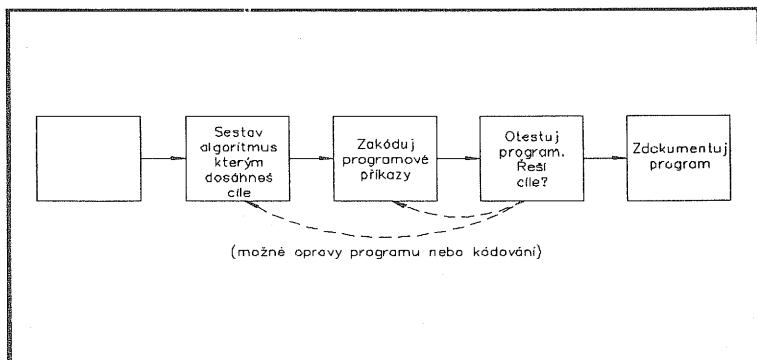
Při navrhování programu pro grafické aplikace můžeme zjistit, že naše programy se nevejdou do paměti (jsou příliš velké), nebo jsou při zpracování příliš pomalé, nebo je příliš obtížná jejich modifikace či ladění. Můžeme očekávat, že se s těmito problémy budeme setkávat tím více, čím jsou naše programy složitější a pokud nevěnujeme dostatečnou pozornost návrhu programu. Použitím standardního postupu návrhu dříve než "skočíme" do kódování, můžeme vytvářet programy, které mají méně problémů a které mohou být snáze upravovány pro různé aplikace. V této kapitole uvedeme některé z technik dobrého návrhu programů a jejich užití v uspořádání našich grafických programů.

## **9.1. Uspořádání grafických programů**

Začneme zkoumáním základních kroků obsažených ve vývoji programů. Pak budeme uvažovat jak ve všech krocích efektivně plánovat a uspořádat naše programy. Dalšími důležitými faktory, které významně ovlivní návrh našich programů, budou pravidla efektivity vyjádřená v nárocích na paměť nebo čas zpracování.

### **9.1.1 Fáze vývoje programů**

V teoreické literatuře se setkáváme s pěti fázemi návrhu libovolného programu (Obr. 11-1). Musíme (1) definovat a načrtout hlavní části, (2) rozhodnout o posloupnosti příkazů (algoritmus) užívaných v tomto programu, (3) napsat skutečné příkazy programu (kódování), (4) testovat program a (5) poskytnout v programu určité vysvětlení (dokumentaci) toho, co jednotlivé části programu dělají. Můžeme si ušetřit čas a lámání hlavy, když se při vývoji našich programů v každém z těchto kroků zamyslíme.



Obr. 11-1 Fáze vývoje programů.

Definování hlavních částí programu znamená, že rozhodneme přesně o tom co chceme, aby program dělal a jak poskytneme data pro vytvoření obrazu. Pokud budeme vytvářet obrázek, definujeme vše, co chceme mít ve všech částech obrázku. Pokud vytváříme graf, rozhodneme o typu grafu, jeho orientaci a jeho umístění na stínítku obrazovky. Data pro vytváření zobrazení by mohla být definována v příkazech DATA, vkládána pomocí příkazů INPUT, počítána z rovnic, nebo libovolnou kombinací uvedených způsobů. Vstupní data mohou být vkládána jednou na začátku programu, nebo může být sestaven tak, aby s námi vedl interaktivní dialog. Specifikací všech vstupních proměnných, přesného formátu použitých dat a hlavních typů požadovaných zpracování (výpočty, třídění, změna měřítka dat na velikost stínítka obrazovky, atd.) zjednoduší naši úlohu vývoje detailního algoritmu a příkazů programu. V tomto bodu také chceme pamatovat na pozdější transformace nebo úpravy, které bychom mohli chtít v našich obrazech dělat, a umožnit to v naší definici programu.

Jakmile máme navrženy cíle programu a definována všechna potřebná data, můžeme vyvíjet uvedené kroky nutné k uskutečnění zpracování každé části programu. To je fáze vývoje algoritmu. Zde můžeme sestavit celkové uspořádání (organizaci) našeho programu podle standardních metod strukturovaného programování. Tyto metody nám umožňují uspořádat naše programy tak, že můžeme tyto programy snadněji testovat a ladit a provádět, je-li to nezbytné, pozdější úpravy.

Metody strukturovaného programování volají, v podstatě, po modulárním návrhu, vyhýbajícím se mnoha větvění, která mohou činit potíže při sledování logiky programu. Sestavíme každý typ zpracování potřebný v programu (vstup dat, změna měřítka dat, transformace, nebo vynášení bodů) jako zvláštní modul. Tyto moduly by mohly být podprogramy nebo pouze zvláštní bloky kódování. Myšlenka, skrytá za vytvářením modulů, tkví v tom, že nesmějeme různé úlohy programu uvnitř jednoho oddílu programu - každý modul plní pouze jednu úlohu. Tudíž máme jeden modul (nebo část programu), který provádí vstup dat, jiný modul, který dejme tomu, třídí data atd. Později bude pro nás snadné určit, co která část programu dělá, budeme-li hledat chybu nebo budeme-li chtít provést změny. Samozřejmě, že každá složka obrázku nebo každá část grafu (osy, sloupce, křivky) mohou být definovány a zobrazovány zvláštním modulem. To nám umožňuje snadné rušení, nebo doplňování částí obrazu. Nakonec moduly uspořádáme do pořadí, ve kterém je chceme zpracovávat, od začátku programu do konce. Moduly, jako jsou vynášecí (kreslící) podprogramy, které jsou využívány ostatními moduly, umístíme na konec programu. Do každého modulu by se mělo vstupovat pouze na jeho začátku a vystupovat pouze na jeho konci. Toto zabraňuje vícenásobným příkazům GOTO a RETURN, které by mohly komplikovat logiku programu. Pořadí modulů můžeme vypsat a ukázat jednotlivé

kroky zpracování pomocí nějaké formy vývojového diagramu.

Kódování skutečných příkazů programu v BASIC (nebo v libovolném jiném jazyku) může být nyní provedeno přímo z našeho návrhu algoritmu, modul po modulu. Pro usnadnění pozdějších možných manipulací s naším zobrazením bychom měli sestavit toto kódování tak, aby všechny zobrazované body byly uloženy v polích. Pak můžeme později snadno do našeho programu doplnit transformační modul, který by mohl zpracovávat body specifikovaného pole. Jména proměnných by měla být volena tak, aby popisovala parametry, které představují a důsledně bychom měli používat stejná jména ve všech programech. Např. můžeme používat označení Q pro množství (Quantity), A pro úhel (angle) a X,Y,Z pro souřadnice. To nám usnadní vysledování kroků zpracování v našich programech, zvláště máme-li napsáno několik různých programů. Odsazovací schéma je velmi užitečné při čtení programů. Pro snadnější identifikaci těla každé smyčky můžeme odsadit všechny příkazy, které se nacházejí mezi příkazy FOR a NEXT. Dále můžeme, pro zlepšení čitelnosti programů, odsadit ostatní části kódu, jako jsou vnořené příkazy IF. Některé systémy automaticky potlačují všechny vodící mezery na příkazovém řádku, což znamená, že odsazování není v tomto případě možné.

Testování programu může být, v zájmu zjednodušení procesu ladění, prováděno po jednotlivých modulech. Není-li to praktické, chceme ještě určit, že každý modul provádí přesně (korektně) zpracovávanou úlohu. Pokud má nás program zobrazovat jeden dlícký obrázek nebo graf, je snadné posoudit zda pracuje správně. Program je korektní, pokud dostaneme obraz, který jsme požadovali. V tomto případě je obvykle snadné určit, co je nutno opravit, pokud nedostaneme ten správný obraz. U programů s obecnějším účelem, jako je program pro kreslení grafu libovolné vstupní křivky, bychom

měli testovat všechny možné logické cesty. To představuje testování různých typů křivek, abychom se přesvědčili, že iato křivka je vždy kreslena ve správné oblasti stínítka obrazovky a že osy a popis jsou správně umístěny. Pokud libovolná část programu nefunguje tak, jak by měla, pak nejprve kontrolujeme, abychom si byli jisti, že kódování odpovídá algoritmu. Může být potřeba opravit pouze kód programu, nebo budeme muset začít ve fázi vývoje algoritmu, abychom jej opravili. Mělo by být použito dostatečné množství testovacích dat, abychom se přesvědčili, že každý modul je korektní.

Dokumentování programu je nepřetržitý proces procházející všemi fázemi vývoje programu. Definice programu, algoritmus a kódovací konvence jsou všechny součástí dokumentace. Mohli bychom sestavit pro naše programy "dokumentační soubor", který by obsahoval definici programu a algoritmu pro naše pozdější odkazy. Dále používání popisných jmen proměnných, odsazování, modulární návrh programu a početné příkazy REM (komentář) mohou učinit naše programy samodokumentovatelné.

### 9.1.2 Omezení efektivity

Dalšími faktory, které mohou ovlivnit uspořádání našich grafických programů jsou hranice velikosti paměti a časové meze. Tato pravidla mohou přímo kolidovat s naším návrhem programu podle metod strukturovaného programování.

Je-li omezující velikost paměti, budeme se muset vzdát určité části čitelnosti programu a dokumentace, abychom ušetřili paměťový prostor. Samozřejmě, mohli bychom program rozdělit tak, že naše grafické zobrazení je vytvářeno a řízeno v částech užívajících více než jeden program.

Program bychom mohli také zkrátit vynecháním opakovaných příkazů nebo částí programu. To by mohlo znamenat kombinování modulů nebo kódových segmentů, které odstraní určitou část modularity. Mezery na řádkách mohou být vynechány, více příkazů může být napsáno na každý řádek a příkazy REM mohou být vynechány. Všechny tyto metody ušetří paměťový prostor, avšak současně způsobí, že program bude obtížněji pochopitelný, laditelný a modifikovatelný.

Časová omezení mohou být splněna s menším vlivem na náš strukturovaný návrh a obecně nejsou obvykle hlavním omezením, pokud neprovádíme animaci obrázku. K urychlení našich programů potřebujeme jednoduše snížit počet výpočtů a zpracování. Pro animaci to znamená přesouvání méně částí a každý pohyb na větší vzdálenost.

Výpočty můžeme redukovat několika způsoby. Všechny nadbytečné výpočty by měly být vynechány. Pokud jsme udělali výpočet jednou, chceme se přesvědčit, že ho neopakujeme. To může někdy nastávat ve smyčkách:

```
610 FOR I = 1 TO N  
620 FOR J = 1 TO 1000  
630 K = I + 3  
.  
.  
.
```

Zde počítáme  $K = I + 3$  tisíckrát pro každou z  $N$  hodnot  $I$ . Příkaz na řádku č. 630 by měl předcházet příkaz na řádku č. 620, aby se zabránilo tomuto typu redundancy. Trigonometrické a exponenciální funkce jsou při výpočtu značně pomalé. Často můžeme zrevidovat náš algoritmus a vyhnout se užití těchto funkcí. Obecně, redukování všech výpočtů pouze na ty, které jsou nezbytné a obsahují pouze sčítání a odčítání celých čísel, urychlí celé zpracování.

Také by bylo možné redukovat výpočty zařazením většího počtu příkazů IF pro vynechání zbytečných výpočtů. Avšak, to zase zvýší nároky na paměť. Vynechání, nebo redukování rozměru polí, může také ušetřit čas. Děle trvá výpočet prvku dourozměrného pole než prvku jednorozměrného pole. Podobně přístup k prvku jednorozměrného pole trvá déle než přístup k jednotlivým proměnným bez indexu. Další metodou zkrácení času výpočtu je psaní našich programů v assembleru nebo v kompliovaném jazyku vysoké úrovni místo interpretovaného jazyka. Některé mikropočítáčové systémy nabízejí speciální grafické funkce, které mohou být užity pro zvýšení rychlosti našich programů. Tyto funkce zahrnují speciální funkce k provádění animací, transformačních operací a vytváření okének (windowing).

## 9.2. Interaktivní metody

---

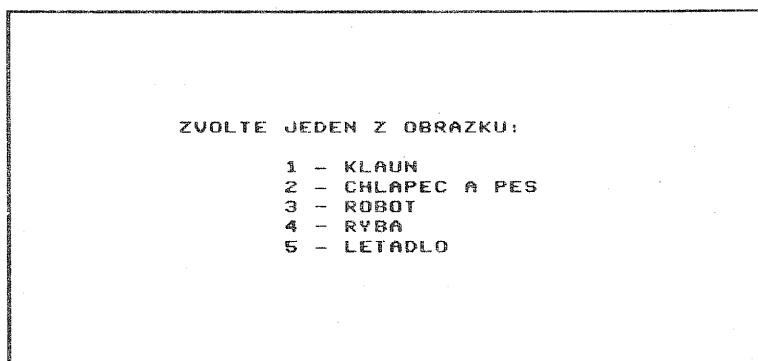
Efektivní metodou vkládání vstupních informací pro mnohé grafické programy je prostřednictvím interaktivního dialogu. Při použití této techniky s námi může program vést, v určitých stádiích zpracování, omezenou konverzaci kladením dotazů, co bychom rádi dále dělali. Naše možnosti mohou být předkládány ve formě nabídky (menu) možností a my můžeme volit jednu z nich pomocí některého ze vstupních interaktivních zařízení. Existuje řada dostupných interaktivních vstupních zařízení. Nejčastěji používaná jsou klávesnice, světlé pero, grafický tablet a paddles.

### 9.2.1 Výběr podle menu

Seznam možností zpracování nabízený nějakým programem je označován jako menu. Mnoho z našich programových příkladů užívalo menu jako prostředek výběru polohy na stínítku obrazovky, transformační metody, která se má použít, nebo části obrázku, která se má transformovat.

Sestavíme-li grafický program, který dovoluje provádět několik možností zpracování, můžeme použít schéma menu k výběru možnosti. Formát menu bude záviset na typu použitého interaktivního vstupního zařízení.

Naše příklady programů předpokládaly, že naše interakce s grafickým programem probíhá pouze prostřednictvím klávesnice. Pro vstup z klávesnice je jednoduchou a efektivní metodou výběr z menu výběr podle číselného nebo abecedního seznamu možností. Napsání číslice nebo písmena podle volby způsobí větvení programu na příslušný modul, který bude uskutečňovat tuto volbu (takovou jako kreslení klauna, napíšeme-li číslo 3). Toto

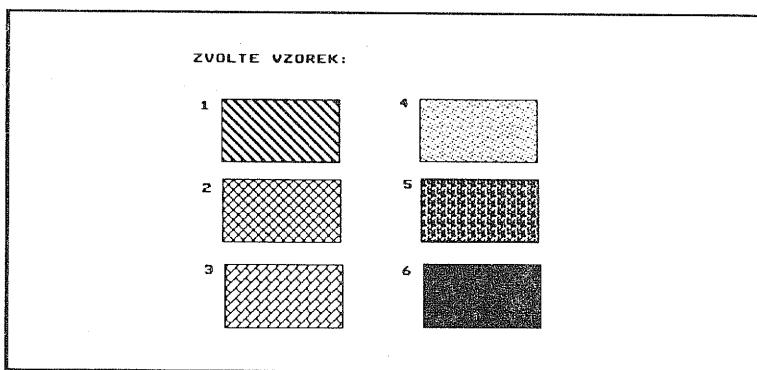


Obr. 11-2 Schéma výběru podle menu s očíslovanými možnostmi.

Schéma výběru můžeme použít pro výpis mnoha různých typů programových možností. Obrázek 11-3 ukazuje výběr stínovacích vzorů pomocí menu a Obr. 11-4 uvádí menu možných výpočtů, které by mohly být prováděny pro určitou množinu vstupních dat. Libovolný znakový řetězec nebo klávesa na klávesnici by mohly být užity k výběru, ale číslování

položek nebo označení písmeny je pro většinu aplikací postačující.

Menu může zaplnit celé stínítko, nebo může být umístěno k jedné straně, nahoru, nebo dolů. Pokud menu vyplní celé stínítko obrazovky, musíme je po provedení výběru vymazat. Umístíme-li však menu do menší části stínítka obrazovky, můžeme zobrazovat obrázek společně s menu, jak je uvedeno na Obr. 11-5. V tomto případě můžeme menu ponechat na stínítku po provedení výběru. To má význam tehdy, pokud jsou nabízené možnosti nabízeny a voleny opakován, protože pak nemusí být menu znova zobrazováno při každém výběru. Po provedení všech výběrů může být menu vymazáno. Současné zobrazování menu a obrázku je také užitečné tehdy, používáme-li jiné typy interaktivních zařízení a chceme-li zobrazovat větší počet menu. Nedostatkem současného zobrazování menu a obrázku nebo grafu je, že se tím zmenší efektivní velikost stínítka obrazovky



Obr. 11-3 Méněný obrazek pro výběr vzorek.

KTERYM ZPUSOBEM CHCETE PROLOZIT KRIUKU?

- K: KUBICKA KRIUKA
- U: KVADRATICKA KRIUKA
- N: NEJMENSI ODCHYLKU
- B: BEZIEROVA KRIUKA

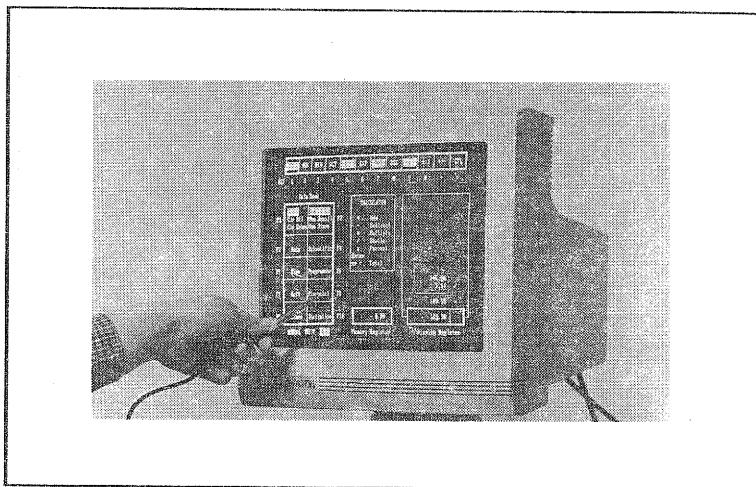
Obr. 11-4 Menu pro výběr možnosti zpracování množiny dat.

1-CLAPEC 2-PES 3-HYDRANT  
PRESUNOUT CAST OBRAZU (0 PRO KONEC)? ■

Obr. 11-5 Menu a obrázek zobrazované současně.

### 9.2.2 Světelná pera

Obrázek 11-6 ukazuje světelné pero při použití k výběru z možností nabízených menu. Světelné pero je zařízení ve tvaru tužky, které může detektovat světlo vyzařované z určitého bodu na stínítku obrazovky. Vyzařované světlo vzniká na žhnoucím fosforovém povlaku stínítka obrazovky při průchodu elektronového paprsku. Pokud je aktivované světelné pero namířeno na bod na stínítku obrazovky, který elektronový paprsek rozsvítíl, jsou souřadnice tohoto bodu uloženy do paměti. Protože elektronový paprsek prochází každým bodem stínítka přibližně 25 krát za sekundu, je detekce rozsvíceného bodu světelným perem prakticky okamžitá.

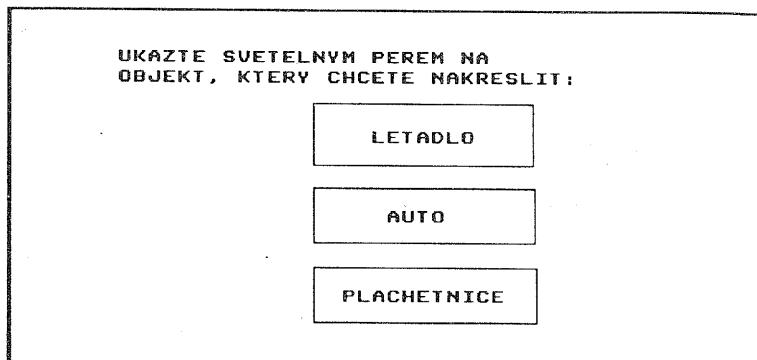


Obr. 11-6 Výběr z menu pomocí světelného pera.

Aktivování světelného pera se provádí různými způsoby. Některá pera mají stisknutelný hrot a jsou aktivována jeho stisknutím proti obrazovce. Jiná mají tlačítko na straně pera, které musíme stisknout. Třetí typ je aktivován dotykem kovové plochy blízko hrotu pera. Příkazy, které mohou být zařazeny do programu v jazyku BASIC k testování aktivace světelného pera, jsou v mnoha mikropočítacích systémech. Je-li programem zjištěno, že světelné pero je "on", může být okamžitá hodnota souřadnic pera na stínítku obrazovky použita pro výběr z menu, pro vynesení bodu nebo kreslení přímky, pro změnu polohy zobrazených objektů atd..

Výběr z menu pomocí světelného pera se v programu provádí testováním souřadnic vrácených perem, aby se určilo, která položka byla vybrána. Vybraná položka je ta, která obsahuje souřadnice pera uvnitř své pravoúhelníkové oblasti. Na Obr. 11-7 předpokládáme, že souřadnice pera leží uvnitř pravoúhlé oblasti s popisem FIRE TRUCK. Testováním hodnoty souřadnice Y můžeme určit, že byla volena možnost FIRE TRUCK. Položky uvedené v menu jako je na Obr. 11-7, by měly být široce odděleny tak, aby mohly být souřadné pozice jasné určeny, že jsou pouze v jedné oblasti. Položky menu mohou být vypsány jako slova, jako barevné pravoúhelníky pro volbu barvy, nebo jako tvary pro volbu objektu (ikony).

Světelné pero můžeme používat také jinak, než k výběru pomocí menu. Můžeme sestavit program k vynášení bodů nebo sérií přímek při pohybu aktivovaného pera po stínítku obrazovky. To nám umožňuje kreslit objekty a obrázky na obrazovce, i když k přesnému zobrazení objektu na obrazovce je nutná pevná ruka. Tentýž program můžeme použít pro digitalizaci obrázku obtahovaného na stínítku obrazovky.

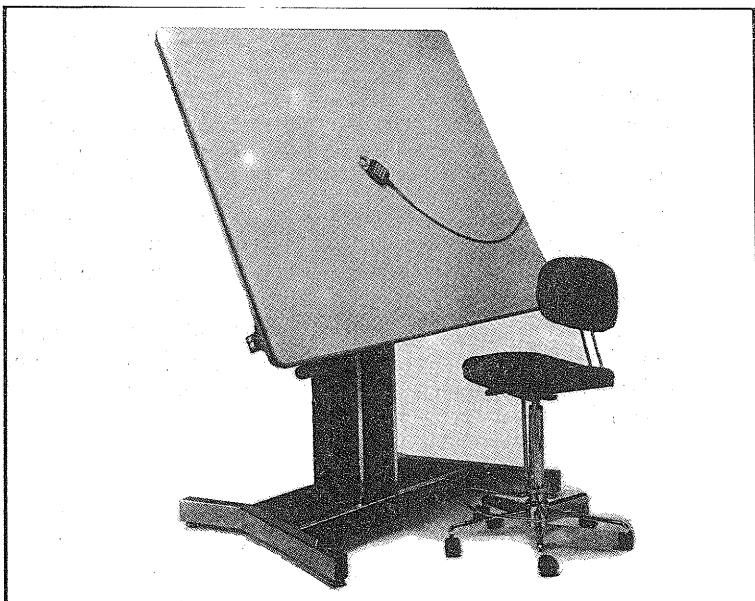


Obr. 11-7 Aktivované světelné pero namířené na slova LETADLO způsobí uložení souřadné pozice uvnitř pravoúhelníka do paměti.

Tento program by pak uložil souřadnice obrázku do souboru, který by později mohl být transformován a kombinován s jinými obrázky nebo texty. Také bychom mohli odvodit programy pro zobrazování libovolného předem určeného tvaru (přímka, pravoúhelník, mnohoúhelník, kružnice) v libovolné poloze volené světelným perem. Translace, změna měřítka, nebo rotace objektů jsou další možné aplikace světelného pera. Můžeme posouvat objekt aktivováním pera na nové pozici stínítka obrazovky. Změna měřítka a rotace mohou být prováděny výběrem podle menu.

### 9.2.3 Grafické digitalizační podložky (tablet)

Podobně jako světelná pera, jsou grafické digitalizační podložky (tablety) zařízení pro výběr souřadných poloh na stínítku obrazovky. Avšak zde, určujeme polohu, místo na stínítku obrazovky, na povrchu digitalizační podložky. Reprezentace souřadnic je na digitalizační podložce shodná jako na stínítku obrazovky s bodem (0,0) v levém horním

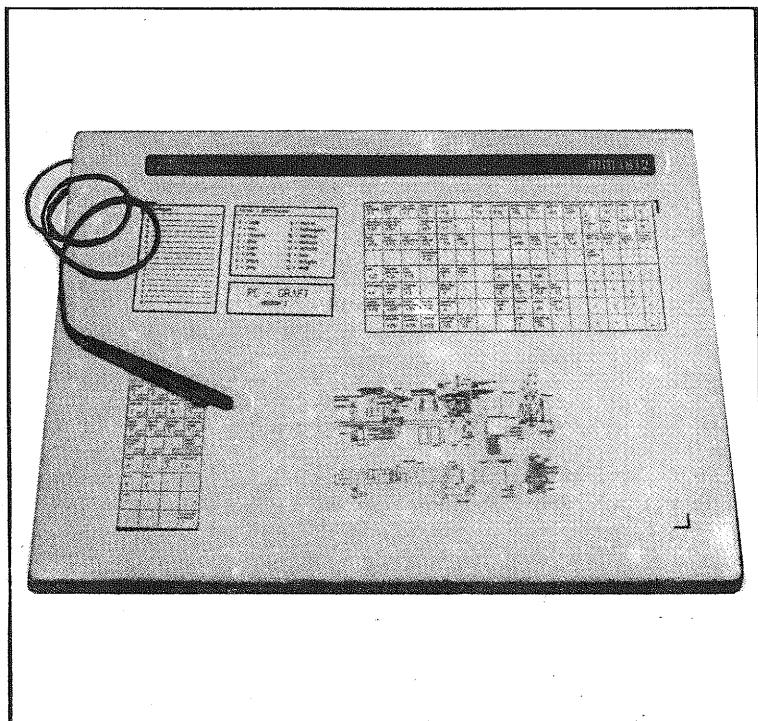


Obr. 11-8 Grafická digitalizační podložka s ručním ukazatelem.

rohu. Souřadnice polohy budou ukládány digitalizační podložkou do paměti, jakmile je aktivovaný ruční ukazatel (kurzor) nebo snímací hrot (stylus). Ruční ukazatel nebo snímací hrot aktivujeme stisknutím tlačítka. Často je k dispozici několik tlačítek, které umožňují snímat jeden bod nebo nepřetržitou řadu bodů při pohybu na digitalizační podložce. Obrázek 11-8 ukazuje digitalizační podložku s ručním ukazatelem a Obr. 11-9 se snímacím hrotom.

Polohu volíme umístěním vlasového křížku na ručním ukazateli s bodem na digitalizační podložce. Digitalizační podložky měří souřadnice pomocí napěťových diferencí na

mřížce vodičů umístěných na povrchu podložky. Na každém vodiči je poněkud odlišné napětí, takže dostáváme napěťové diference, které ve všech směrech odpovídají diferencím souřadnic. Aktivací snímacího hrotu nebo ručního ukazatele určíme, že napětí této polohy mají být dále vyhodnocena. Tato napětí jsou převedena na hodnoty souřadnic X, Y a zaznamenána do paměti. Některé digitalizační podložky používají k určování souřadnic místo napětí zvuk. Tyto zvukové digitalizační podložky obsahují malé páskové mikrofony umístěné podél okrajů podložky. Snímací hrot při své aktivaci vytváří jiskru, jejíž zvuk se šíří směrem k okrajům podložky. Časy, které tento zvuk potřebuje k dosažení mikrofonů ve směrech X a Y jsou převedeny na hodnoty souřadnic.



Obr. 11-9 Grafická digitalizační podložka se snímacím hrוטem (stylus).

K načtení hodnot souřadnic z digitalizační podložky do paměti počítače se obvykle používá program psaný v assembleru. Tento program je pak volán z programu v jazyce BASIC pomocí příkazu CALL. Tyto programy budou odlišné pro různé systémy a přo různé typy digitalizačních podložek.

Použití digitalizační podložky v interaktivní grafice jsou podobná jako u světelného pera. Digitalizační podložka může

zaznamenávat souřadnice polohy mnohem přesněji, než světelné pero, takže je to dobré zařízení pro přenášení grafů, diagramů, nebo obrázků na stínítko obrazovky. Souřadnice polohy ze zobrazení pak mohou být uchovány ve speciálních souborech pro pozdější použití. Poloha objektů pro translaci je opět určována volbou výsledné polohy na povrchu podložky. Také můžeme na stínítku obrazovky vytvářet obrazce kreslením čar, nebo použijeme podložku k výběru podle menu. Výběrem souřadnic polohy uvnitř oblasti obsazené dílčím tvarem menu je volena konkrétní nabídka. Podobně můžeme navrhnut šablony pro práci s programy, které volí barvy nebo možnosti zpracování.

#### 9.2.4 Analogové pákové ovladače - joysticky

Existuje řada pákových ovladačů, které mohou být připojeny k mikropočítači. Obvykle tento ovladač obsahuje "kniplo", který může být přesouván dopředu a dozadu, nahoru a dolů. Na většině systémů jsou k dispozici příkazy, sloužící k převodu pohybu tohoto kniplu na souřadnicové informace, které mohou být vkládány do programu v jazyku BASIC. Při přesunu kniplu do určitého směru vyhodnocovací elektronika vytváří souřadnice pro tento směr. Takto můžeme pohybem kniplu přesouvat objekty plynule po stínítku obrazovky. Pohyb kniplu může být také v našich programech převeden na změnu měřítka objektu, nebo na rotaci. Na některých ovladačích jsou k dispozici také tlačítka nebo vypínače, takže mohou být aktivovány podobně jako světelná pera a digitalizační podložky. Tudíž můžeme užívat souřadnice vytvářené aktivací ovladače k výběru pomocí menu nebo ke kreslení tvarů.

#### 9.2.5 Samostatné elektronické kursory - myši

V současné době většina grafických systémů využívá

elektronický kurzor - myš. Myš, v podobě samostatné krabičky ovládané pohybem ruky, je optimálním zařízením pro výběr jednotlivých nabídek z menu na obrazovce. Jestliže jsme předchozí grafická zařízení - světlá pera, grafické digitalizační podložky a analogové pákové ovladače - představovali orientačně v případě myší s potěšením můžeme konstatovat, že agilní 602. ZO Svazarmu organizuje Kurs elektronické myší. Zde ve třech manuálech mají všichni jeho účastníci poznat všechny teoretické zákonitosti práce s elektronickou myší. Kurs je doplněn o stavebnici jednoduché, avšak plně funkční elektronické myší s vynikajícím stavebním návodom a potřebným programovým vybavením. Tak za cenu necelých pětiset korun máte možnost nejen dokonale poznat teoretické základnosti interaktivních grafických zařízení, ale vlastnit i plnohodnotnou elektronickou myš. Na závěr ještě adresa pro případné zájemce o Kurs elektronická myš - 602. ZO Svazarmu, Winterova 14, 16000 Praha 6.

### **9.3. Konstruování obrázku**

---

Interaktivní techniky lze efektivně použít při konstruování obrázků po částech. Takové techniky jsme již použili elementárním způsobem v Prog. 3-18, kde jsme konstruovali obrázek po jednotlivých úsečkách. Metoda postupné konstrukce obrázku může být nyní rozšířena tak, aby obsahovala další tvary a transformace těchto tvarů. Touto metodou můžeme sestavit určitý počet základních "stavebních bloků" a sestavovat obrázek použitím těchto předem definovaných tvarů. Můžeme interaktivně, zvolenými transformacemi, přesouvat libovolný z těchto tvarů do polohy uvnitř našeho obrázku. Program 11-1 uvádí metodu konstruování obrázku z úseček, trojúhelníků, pravoúhelníků a kružnic. Tento program zobrazuje menu, nabízející všechny tvary, na levém okraji stínítka obrazovky. Další menu, pro volbu transformací, je uvedeno na dolním okraji stínítka.

Blikajícím slovem SELECT program označuje které z obou menu máme použít. Obrázek 11-10 ukazuje zobrazení obou menu na stínátku obrazovky

```
10 'PROGRAM 11-1. VYTVAZENI OBRAZKU ZE SOUCASTI.
20   'ZOBRAZUJE DVE MENU NA STINITKU UMOZNUJICI
30   'VOLBU TVARU A TRANSFORMACI. PRO KAZDY VYBRANY
40   'TVAR JE MOZNO VOLIT LIBOVOLNY POSET A KOMBINACI
50   'TRANSFORMACI.
60   ****
70 CLS
80 YA = 5/12  'YA A XA JSOU KOREKCE ROZLISOVACI SCHOPNOSTI
90 XA = 12/5
100 XM = 639
110 YM = 199
120 SCREEN 2
130 GOSUB 830  'TVORBA OBOU MENU
140 AA = 0      'AA JE SOUCTOVY UHEL NATOCENI KRUZNICE
150   **** VSTUPNI VOLBY ****
160 LOCATE 1,1
170 PRINT "      "; 'BLIKANI "SELECT" TVARU
180 FOR K = 1 TO 300:NEXT K
190 LOCATE 1,1
200 PRINT "SELECT:";
210 FOR K = 1 TO 300:NEXT K
220 A$ = INKEY$  'A$ JE VYBER TVARU
230 IF A$ = "" OR A$ < "1" OR A$ > "4" THEN 160
240 RO = VAL(A$) * 4
250 N = 0          'N INDIKUJE ZDA STAVAJICI ZOBRAZENI TVARU
260 LOCATE 23,1    'BY MELO BYT VYMAZANO NEBO NE
270 PRINT "SELECT:"; 'BLIKANI CISLA TVARU A "SELECT"
280 LOCATE RO,1    'TRANSFORMACE
290 PRINT "      ";
300 FOR K = 1 TO 300:NEXT K
310 LOCATE 23,1
320 PRINT "      ";
330 LOCATE RO,1
340 PRINT A$;
350 FOR K = 1 TO 300:NEXT K
360 B$ = INKEY$  'B$ JE VOLBA TRANSFORMACE
```

```

370 IF B$ = "" OR B$<>"T" AND B$<>"S" AND B$<>"R" AND
    B$<>"E" AND B$<>"N" AND B$<>"Q" THEN 260
380 IF B$ = "Q" THEN 3110
390 IF B$ <> "N" THEN 440
400 RESTORE 'NAVRAT NA ZACATEK DAT
410 GOSUB 990
420 '***** VYMAZAT ? *****
430 GOTO 140
440 IF B$ <> "E" THEN 530
450 C=0 'NASTAVENI BARVY POPREDI NA BARVU POZADI
460 IF B$ = "E" AND A$ = "1" THEN GOSUB 1070
470 IF B$ = "E" AND A$ = "2" THEN GOSUB 1100
480 IF B$ = "E" AND A$ = "3" THEN GOSUB 1160
490 IF B$ = "E" AND A$ = "4" THEN GOSUB 1430
500 RESTORE
510 GOSUB 990
520 GOTO 140
530 LOCATE 23,1: PRINT "      ";
540 LOCATE 24,1: PRINT "      ";
550 '***** TRANSLACE ? *****
560 IF B$ <> "T" THEN 640
570 GOSUB 1850 'TISK INSTRUKCI PRO TRANSLACI
580 C=0 'NASTAVENI BARVY POPREDI NA BARVU POZADI
590 IF B$ = "T" AND A$ = "1" THEN GOSUB 1510
600 IF B$ = "T" AND A$ = "2" THEN GOSUB 1590
610 IF B$ = "T" AND A$ = "3" THEN GOSUB 1690
620 IF B$ = "T" AND A$ = "4" THEN GOSUB 1760
630 '***** ZMENA MERITKA ? *****
640 IF B$ <> "S" THEN 720
650 GOSUB 2330 'TISK INSTRUKCI PRO MERITKO
660 C=0 'NASTAVENI BARVY POPREDI NA BARVU POZADI
670 IF B$ = "S" AND A$ = "1" THEN GOSUB 1920
680 IF B$ = "S" AND A$ = "2" THEN GOSUB 2020
690 IF B$ = "S" AND A$ = "3" THEN GOSUB 2140
700 IF B$ = "S" AND A$ = "4" THEN GOSUB 2220
710 '***** ROTACE ? *****
720 IF B$ <> "R" THEN 790
730 GOSUB 3000 'TISK INSTRUKCI PRO ROTACI
740 C=0 'NASTAVENI BARVY POPREDI NA BARVU POZADI
750 IF B$ = "R" AND A$ = "1" THEN GOSUB 2400
760 IF B$ = "R" AND A$ = "2" THEN GOSUB 2540

```

```

770 IF B$ = "R" AND A$ = "3" THEN GOSUB 2740
780 IF B$ = "R" AND A$ = "4" THEN GOSUB 2830
790 N = 1      'AKTIVACE VYMAZANI STAREHO OBRAZU TVARU
800 GOSUB 960 'NOVE ZOBRAZENI MENU
810 GOTO 260
820 '
830 ##### ZOBRAZENI MENU #####
840 LOCATE 1,1: PRINT "SELECT:";
850 LOCATE 4,1: PRINT "1.";
860 LOCATE 8,1: PRINT "2.";
870 LOCATE 12,1: PRINT "3.";
880 LOCATE 16,1: PRINT "4.";
890 C=1
900 GOSUB 990
910 GOSUB 1080
920 GOSUB 1100
930 AA = 0
940 GOSUB 1160
950 GOSUB 1430
960 LOCATE 23,1: PRINT "SELECT:";
970 LOCATE 24,1: PRINT "T -TRANSLATE   S -SCALE   R
-ROTATE   E -ERASE   N -NEXT OBJECT   Q -QUIT";
980 RETURN
990 ***** NACTENI DAT BODU *****
1000 READ XP,YP,XQ,YQ
1010 READ XU,YU,XV,YV,XW,YW,XX,YX
1020 READ XC,YC,RX,RY
1030 READ XR,YR,XS,YS,XT,YT
1040 RETURN
1050 '
1060 ##### KRESLICI PODPROGRAMY #####
1070 ***** KRESLENI USECKY *****
1080 LINE(XP,YP)-(XQ,YQ),C
1090 RETURN
1100 ***** KRESLENI OBDELDNIKA *****
1110 LINE(XU,YU)-(XV,YV),C
1120 LINE(XV,YV)-(XW,YW),C
1130 LINE(XW,YW)-(XX,YX),C
1140 LINE(XX,YX)-(XU,YU),C
1150 RETURN
1160 ***** KRESLENI KRUZNICE *****

```

```

1170 IF RX < RY THEN 1200
1180 R = RY
1190 GOTO 1210
1200 R = RX
1210 IF AA <> 0 THEN 1310    'TVAR BYL NATOCEN. UZIT
    ALTERNATIVNI RUTINU
1220 FOR A = 0 TO 1.5708 STEP 1/R  'VYNESENI 4
    SYMETRICKYCH BODU
1230     DX = RX * COS(A)
1240     DY = RY * SIN(A)
1250     PSET(XC+DX, YC+DY*YA), C
1260     PSET(XC+DX, YC-DY*YA), C
1270     PSET(XC-DX, YC+DY*YA), C
1280     PSET(XC-DX, YC-DY*YA), C
1290 NEXT A
1300 GOTO 1420
1310 CX = SIN(AA) * XA  'VYPOCET KONSTANTNICH CASTI ROVNICE
1320 CY = SIN(AA) * YA
1330 FOR A = 0 TO 3.14159 STEP 1/R
1340     X = RX * COS(A)          'ROVNICE KRUZNICE
1350     Y = RY * SIN(A) * YA
1360     YH = X
1370     X = X * COS(AA) + Y * CX  'ROVNICE ROTACE
1380     Y = Y * COS(AA) - XH * CY
1390     PSET(XC+X, YC+Y), C  'VYNESENI 2 SYMETRICKYCH BODU
1400     PSET(XC-X, YC-Y), C
1410 NEXT A
1420 RETURN
1430 ***** KRESLENI TROJUHELNika *****
1440 LINE(XR, YR)-(XS, YS), C
1450 LINE(XS, YS)-(XT, YT), C
1460 LINE(XT, YT)-(XR, YR), C
1470 RETURN
1480 '
1490 ##### TRANSLACE #####
1500 ***** TRANSLACE USECKY *****
1510 IF N = 0 THEN 1530  'JE-LI N=0, JE TVAR STALE V MENU
    NEMAZAT
1520 GOSUB 1070  'MAZANI
1530 XP = XP + HT: YP = YP + VT
1540 XQ = XQ + HT: YQ = YQ + VT

```

```

1550 C=1
1560 GOSUB 1070      'KRESLENI
1570 RETURN
1580      ***** TRANSLACE OBDELNIKA *****
1590 IF N = 0 THEN 1610
1600 GOSUB 1100      'MAZANI
1610 XU = XU + HT: YU = YU + VT
1620 XV = XV + HT: YV = YV + VT
1630 XW = XW + HT: YW = YW + VT
1640 XX = XX + HT: YX = YX + VT
1650 C=1
1660 GOSUB 1100      'KRESLENI
1670 RETURN
1680      ***** TRANSLACE KRUZNICE *****
1690 IF N = 0 THEN 1710
1700 GOSUB 1160      'MAZANI
1710 XC = XC + HT: YC = YC + VT
1720 C=1
1730 GOSUB 1160      'KRESLENI
1740 RETURN
1750      ***** TRANSLACE TROJUHELNÍKA *****
1760 IF N = 0 THEN 1780
1770 GOSUB 1430      'MAZANI
1780 XR = XR + HT: YR = YR + VT
1790 XS = XS + HT: YS = YS + VT
1800 XT = XT + HT: YT = YT + VT
1810 C=1
1820 GOSUB 1430      'KRESLENI
1830 RETURN
1840      ***** INSTRUKCE *****
1850 LOCATE 23,1: PRINT "          ";
1860 LOCATE 23,1
1870 PRINT "HORIZ. A VERTIKALNÍ VZDALENOST TRANSLACE";
1880 INPUT HT, VT
1890 RETURN
1900 '
1910 ##### ZMENA MERITKA #####
1920      ***** MERITKO USECKY *****
1930 IF N = 0 THEN 1530
1940 GOSUB 1070      'MAZANI
1950 XF = (XP + XQ) / 2

```

```

1960 YF = (YP + YQ) / 2
1970 XP = XP*HS + XF*(1 - HS): YP = YP*VS + YF*(1 - VS)
1980 XQ = XQ*HS + XF*(1 - HS): YQ = YQ*VS + YF*(1 - VS)
1990 C=1
2000 GOSUB 1070    'KRESLENI
2010 RETURN
2020      ***** MERITKO OBDELNIKA *****
2030 IF N = 0 THEN 2050
2040 GOSUB 1100    'MAZANI
2050 XF = (XU + XV + XW + XX) / 4
2060 YF = (YU + YV + YW + YX) / 4
2070 XU = XU*HS + XF*(1 - HS): YU = YU*VS + YF*(1 - VS)
2080 XV = XV*HS + XF*(1 - HS): YV = YV*VS + YF*(1 - VS)
2090 XW = XW*HS + XF*(1 - HS): YW = YW*VS + YF*(1 - VS)
2100 XX = XX*HS + XF*(1 - HS): YX = YX*VS + YF*(1 - VS)
2110 C=1
2120 GOSUB 1100    'KRESLENI
2130 RETURN
2140      ***** MERITKO KRUZNICE *****
2150 IF N = 0 THEN 2170
2160 GOSUB 1160    'MAZANI
2170 RX = RX * HS
2180 RY = RY * VS
2190 C=1
2200 GO$UB 1160    'KRESLENI
2210 RETURN
2220      ***** MERITKO TROJUHELNIKA *****
2230 IF N = 0 THEN 2250
2240 GOSUB 1430    'MAZANI
2250 XF = (XR + XS + XT) / 3
2260 YF = (YR + YS + YT) / 3
2270 XR = XR*HS + XF*(1 - HS): YR = YR*VS + YF*(1 - VS)
2280 XS = XS*HS + XF*(1 - HS): YS = YS*VS + YF*(1 - VS)
2290 XT = XT*HS + XF*(1 - HS): YT = YT*VS + YF*(1 - VS)
2300 C=1
2310 GOSUB 1430    'KRESLENI
2320 RETURN
2330      ***** INSTRUKCE *****
2340 LOCATE 23,1
2350 PRINT "UVED MERITKA X A Y";
2360 INPUT HS, VS

```

```

2370 RETURN
2380 '
2390 '#####
2400 '***** ROTACE #####
2410 IF N = 0 THEN 2430
2420 GOSUB 1070 'MAZANI
2430 XO = (XP + XQ) / 2 'URCENI STREDU USECKY
2440 YO = (YP + YQ) / 2
2450 XH = XP 'USCHOVANI HODNOTY XP PRO VYPOCTY Y
2460 XP = XO + (XP-XO) * COS(AR) + (YP-YO) * SIN(AR) * XA
2470 YP = YO + (YP-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2480 XH = XQ
2490 XQ = XO + (XQ-XO) * COS(AR) + (YQ-YO) * SIN(AR) * XA
2500 YQ = YO + (YQ-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2510 C=1
2520 GOSUB 1070 'KRESLENI
2530 RETURN
2540 '***** ROTACE OBDELNIKA *****
2550 IF N = 0 THEN 2570
2560 GOSUB 1100 'MAZANI
2570 XO = (XU + XV + XW + XX) / 4 'URCENI STREDU OBDELNIKA
2580 YO = (YU + YV + YW + YX) / 4
2590 XH = XU 'USCHOVANI OKAMZITE HODNOTY XU
2600 XU = XO + (XU-XO) * COS(AR) + (YU-YO) * SIN(AR) * XA
2610 YU = YO + (YU-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2620 XH = XV
2630 XV = XO + (XV-XO) * COS(AR) + (YV-YO) * SIN(AR) * XA
2640 YV = YO + (YV-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2650 XH = XW
2660 XW = XO + (XW-XO) * COS(AR) + (YW-YO) * SIN(AR) * XA
2670 YW = YO + (YW-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2680 XH = XX
2690 XX = XO + (XX-XO) * COS(AR) + (YX-YO) * SIN(AR) * XA
2700 YX = YO + (YX-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2710 C=1
2720 GOSUB 1100 'KRESLENI
2730 RETURN
2740 '***** ROTACE KRUZNICE *****
2750 IF N = 0 THEN 2780
2760 AA = AS
2770 GOSUB 1160 'MAZANI

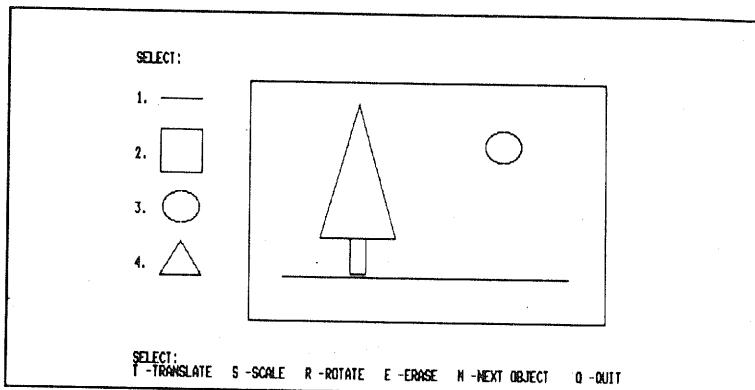
```

```

2780 AA = AR + AS
2790 C=1
2800 GOSUB 1160    'KRESLENI
2820 RETURN
2830      ***** ROTACE TROJUHELNIKA *****
2840 IF N = 0 THEN 2860
2850 GOSUB 1430    'MAZANI
2860 XO = (XR + XS + XT) / 3  'URCENI STREDU TROJUHELNIKA
2870 YO = (YR + YS + YT) / 3
2880 XH = XR
2890 XR = XO + (XR-XO) * COS(AR) + (YR-YO) * SIN(AR) * XA
2900 YR = YO + (YR-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2910 XH = XS
2920 XS = XO + (XS-XO) * COS(AR) + (YS-YO) * SIN(AR) * XA
2930 YS = YO + (YS-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2940 XH = XT
2950 XT = XO + (XT-XO) * COS(AR) + (YT-YO) * SIN(AR) * XA
2960 YT = YO + (YT-YO) * COS(AR) - (XH-XO) * SIN(AR) * YA
2970 C=1
2980 GOSUB 1430    'KRESLENI
2990 RETURN
3000      ***** INSTRUKCE *****
3010 LOCATE 23,1
3020 PRINT "VELIKOST NATOCENI VE STUPNICH";
3030 INPUT AR
3040 AR = AR * 3.14159 / 180  'PREVOD AR NA RADIANY
3050 RETURN
3060 '# ##### #####
3070 DATA 30,27,80,27
3080 DATA 80,45,80,70,30,70,30,45
3090 DATA 55,90,22,22
3100 DATA 30,130,80,130,55,110
3110 END

```

Prog. 11-1 Interaktivní návrh obrázku použitím základních tvarů (úsečka, trojúhelník, obdélník, kružnice) volených z menu.

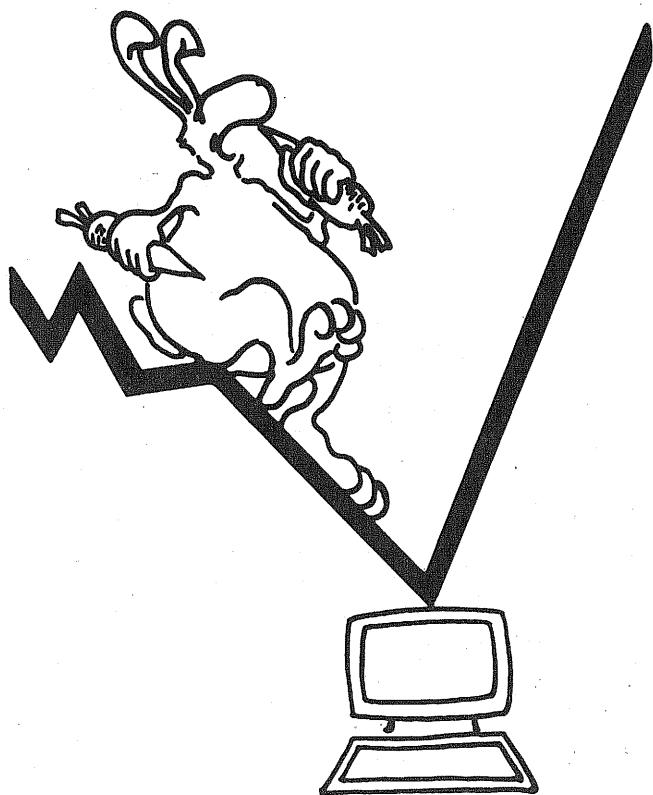


Obr. 11-10 Zobrazení dvou menu a obrázku vytvořeného programem Prog. 11-1.

Program 11-1 je konstruován z modulů, které provádějí jednotlivé funkce programu. Výběr podle menu je prováděn prostřednictvím klávesnice. Tento program můžeme upravit, např. pro světelné pero či myš, doplněním příkazů testujících aktivaci pera nebo myši. Dále je třeba testovat odpovídající vstup souřadnic tak, aby se určilo, která z položek menu má být zvolena. Mohli bychom volit vzdálenost posuvu, měřítka, nebo úhly natočení pomocí třetího menu. Toto menu může být jednoduše úsečka (nebo řada úseček) se stupnicí představující velikost. Pro výběr úhlu natočení může být příslušná úsečka dělena od 0 do 360 stupňů. Pro vzdálenosti posuvu by byla úsečka dělena od záporných po kladné hodnoty v rozsahu možných hodnot souřadnice.

Vytváření obrázků postupnou metodou, jako v Prog. 11-1, je obecnou metodou používanou v mnoha aplikačních oblastech. Tuto techniku bychom mohli použít v aplikacích navrhování, vynášení grafů, uměleckých vzorů, hrách, nebo pro experimentování s tvary.

## 10. PODNIKOVÁ GRAFIKA

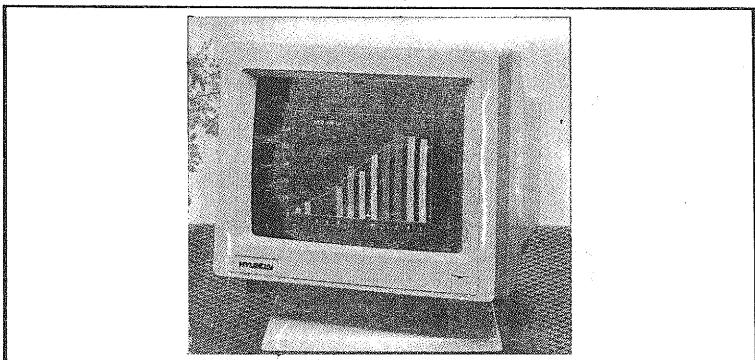


Podnikové využití počítačové grafiky je jednou z nejrozsáhlejších a nejvíce se vyvíjejících aplikačních oblastí. Počítačem vytvářené grafy, diagramy a obrázky jsou všeobecně užívány jako pomůcky pro finanční analýzy, studie trhu, při plánování či rozhodování. To jsou typické funkce potřebné pro práci u mnoha organizací. Můžeme vytvářet grafy a diagramy poskytující informace o rozpočtu, inventáři, pohybu hotovosti, čistém příjmu, úročích, návratu investic, nebo investiční analýze. Můžeme vynášet grafy pro srovnání cen nebo výrobních charakteristik mezi konkurenty, ukazující regionální zvyky nakupování, nebo trendy odbytu vynášené podle oblasti, prodavače, nebo roku. Grafy demografických dat mohou být užitečné pro vyhledávání potenciálních zákazníků nebo úpravy prodejních teritorií. Síťové grafy a časové diagramy poskytují pomoc při řízení projektů. Obrazové výstupy mohou pomáhat při plánování rozmístění zařízení a vybavení.

Můžeme sestavovat podnikové grafy a diagramy tak, že znázorňují data podle zeměpisné oblasti, nebo podle oddělení organizace. Můžeme také graficky porovnávat nová data se starými v zájmu poskytnutí rychlého srovnání nebo pro indikaci budoucích trendů. Takové grafy jsou používány ve vnitřních zprávách, zprávách o stavu pro zákazníky, nebo v prezentacích. V této kapitole budeme diskutovat některé z metod vytváření různých typů podnikových grafů a diagramů.

### **10.1. Obecné techniky**

Metody vynášení grafů byly uvedeny v předchozích kapitolách. Nyní uvažujme některá rozšíření těchto základních metod, která jsou všeobecně užívána v podnikové grafice.



Obr. 12-1 Počítačová grafika může pomoci zkrátit časové trvání porad rychlejší prezentací koncepcí a vzájemných vztahů mezi daty.

Všeobecnou technikou, užívanou pro diagramy kruhových výsečí (koláčové diagramy), je zvýraznění jedné nebo více částí posunutím radiálně směrem od středu. Příklad takového roztrženého diagramu je uveden na Obr. 12-2. Technika, použitá v Prog. 12-1 k vytvoření tohoto diagramu, je založena na metodě, kterou již jsme diskutovali v předchozích částech při umísťování popisů na diagramy kruhových výsečí. Nejprve určíme úhlové hranice části, která má být odtržena. Pak vypočítáme úhel osy této výseče ve směru poloměru. Nakonec najdeme střední bod pro odtrženou část mimo střed diagramu, ležící na této ose, a nakreslíme celou část z tohoto bodu. Program 12-1 odděluje části posunem o jednu pětinu poloměru diagramu.

```
10 'PROGRAM 12-1. ROZTRZENY KOLACOVY DIAGRAM. KOLACOVY
20      'DIAGRAM S POPISEM A S ODTRZENOU LIBOVOLNOU CASTI.
30 CLS
40 DIM N$(8), V(8)
50 PRINT "UVED MAXIMALNI HORIZONTALNI A VERTIKALNI"
60 PRINT "HODNOTY PRO TENTO ROZLISOVACI REZIM"
```

```

70 INPUT XM, YM
80 PRINT "UVED POSET ZNAKU, KTERE MOHOU TISKNUTY"
90 PRINT "PRES STINITKO OBRAZOVKY V TOMTO"
100 PRINT "ROZLISOVACIM REZIMU"
110 INPUT C1
120 PC = XM / C1 'PC JE POSET BODU NA ZNAK HORIZONTALNE
130 PRINT "UVED POSET MOZNYCH TISKOVYCH RADEK"
140 PRINT "MOZNYCH V TOMTO ROZLISOVACIM REZIMU"
150 INPUT R1
160 PR = YM / R1 'PR JE POSET BODU NA ZNAK VERTIKALNE
170 PRINT
180 PRINT "UVED SOURADNICE STREDU DIAGRAMU"
190 INPUT XN, YN
200 PRINT "UVED POLOMER"
210 INPUT R
220 IF XN+R > XM OR XN-R < 0 OR YN+R > YM OR YN-R < 0 THEN
    1020
230 PRINT
240 '***** VSTUP DAT *****
250 PRINT "UVED POSET ODDILO (MAX. 8)"
260 INPUT N
270 PRINT "UVED JMENO A HODNOTU KAZDEHO ODDILU"
280 T = 0
290 'VSTUP DAT. URCENI SOUCTU (T) VSECH HODNOT.
300 FOR K = 1 TO N
310     INPUT N$(K), V(K)
320     T = T + V(K)
330 NEXT K
340 PRINT "ODTRHNOUT KTERY ODDIL (1 -";N; ")"
350 INPUT E
360 IF E < 1 OR E > N THEN 340
370 '***** VYTVARANI KOLACOVEHO DIAGRAMU *****
380 CLS
390 SCREEN 1
400 YA = 5 / 6 'YA JE KOREKCE ROZLISOVACI SCHOPNOSTI
410 B = 0 'B JE UHEL, KTERY URCUJE PREDCHOZI RADEK
420 S = 0
430 RE = 360 * 3.14159 / 180 'RE JE RADIANOVY EKVIVALENT
    360 STUPNU
440 FOR K = 1 TO N
450     XC = XN 'XN JE, YN JE STRED PRO NORMALNI CAST

```

```

460      YC = YN
470          'VYNASENA PRIMKA JE URCENA HODNOTOU TOHOTO
480          'ODDILU PLUS PREDCHAZEJICICH ODDILU
490      S = S + V(K)
500          'UHEL, KTERY SE MA POUZIT, JE URCEN
510          'PROCENTEM S / T KRUHU
520      A = RE * S / T
530      IF K <> E THEN 660  'MA BYT TENTO ODDIL ODTRZEN ?
540      AC = B + (A-B) / 2  'AC JE UHEL PRIMKY PULICI ODDIL
550      XE = XC + R/5 * COS(AC)  'ANO, TAK URCIT NOVY
      STRED (XE,YE)  1/5 VNE OD
560      YE = YC + R/5 * SIN(AC) * YA  'NORMALNIHO STREDU.
570      XP = XE + R * COS(B)  'PAK NAJIT KONCOVE BODY A
580      YP = YE + R * SIN(B) * YA  'NAKRESLIT.
590      LINE(XE,YE)-(XP,YP)
600      XP = XC + R * COS(A)
610      YP = YC + R * SIN(A) * YA
620      LINE(XC,YC)-(XP,YP)
630      XC = XE
640      YC = YE
650
660      FOR A1 = B TO A STEP 1/R
670          XP = XC + R * COS(A1)
680          YP = YC + R * SIN(A1) * YA
690          PSET(XP,YP)
700      NEXT A1
710      LINE(XC,YC)-(XP,YP)
720      'NATISKNOUT POPIS ODDILU
730          'NAJIT BOD 4 JEDNOTKY DALE SMEREM OD STREDU
740          'MIMO OBLOUK PATRICI K ODLU
750      AC = B + (A - B) / 2  'AC JE UHEL OD HORIZONTALY
760      XL = XC + (R + 4) * COS(AC)  'KTERA BUDE PULIT
      ODDIL
770      YL = YC + (R + 4) * SIN(AC) * YA
780          '(XL,YL) JE BOD UZITY K UMISTENI POPISU
790          'UZIT TENTO BOD JAKO ZACATEK POPISU, JE-LI
800          'NA PRAVE STRANE KRUZNICE. JAKO KONEC POPISU,
810          'JE-LI VLEVO. JAKO STRED, JE-LI NAD NEBO POD
820          'KRUZNICI
830      IF XL > XC + 10 THEN 930
840          'BOD JE KONCEM POPISU

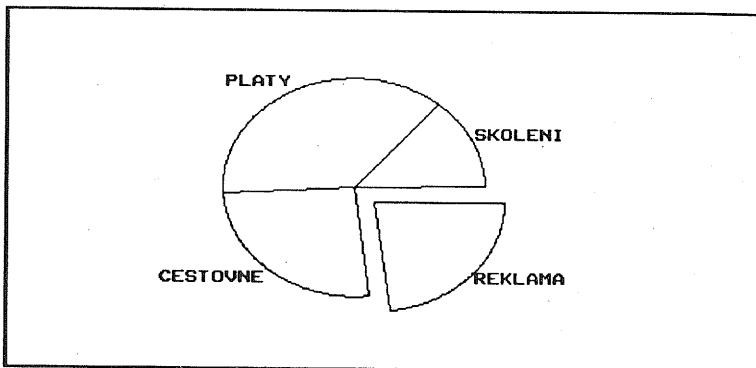
```

```

850      IF XL < XC - 10 THEN 900
860      'JINAK JE BOD STREDEM POPISU. KORIGUJ XL POMOCI
870      '1/2 POCTU BODU POTREBNYCH PRO POPIS
880      XL = XL - LEN(N$(K)) / 2 * PC
890      GOTO 930
900      'BOD JE KONCEM POPISU. PRESUNOUT ZPET O POCET
910      'BODU POTREBNYCH PRO POPIS
920      XL = XL - LEN(N$(K)) * PC
930      'PREVOD SOURADNIC V BODECH (XL,YL) NA CO NEJ-
940      'BLIZSII ODPOVIDAJICI TISKOVE POZICI
950      RO = INT(YL / PR)+1
960      CO = INT(XL / PC)+1
970      LOCATE RO,CO
980      PRINT N$(K);
990      B = A
1000 NEXT K
1010 GOTO 1030
1020 PRINT "SOURADNICE MIMO ROZSAH"
1030 END

```

Prog. 12-1 Roztržený diagram kruhových výsečí (koláčový diagram).



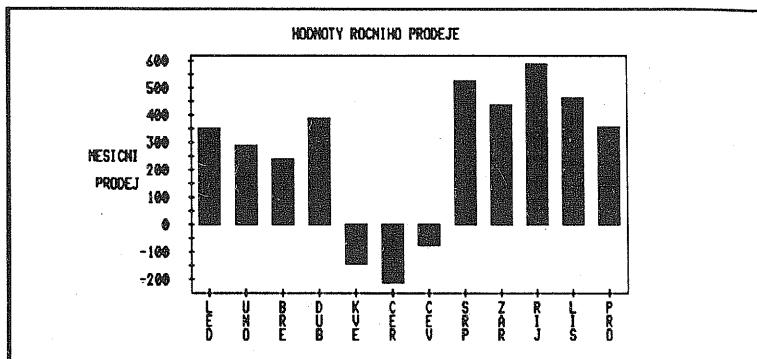
Obr. 12-2 Roztržený diagram kruhových výsečí, zobrazený programem Prog. 12-1.

Někdy bychom však rádi vytvořili grafy, ukazující negativní hodnoty, jako jsou finanční ztráty. To můžeme udělat

vynášením negativních hodnot pod horizontální osu, jak je uvedeno na Obr. 12-3. Tyto grafy jsou vytvořeny jednoduchým prodloužením vertikální osy pod horizontální osu a popisem jak pozitivních, tak negativních hodnot.

Vynášení dvou typů grafů na tutéž osu je efektivním prostředkem kondenzace diagramů nebo porovnávání dat. Program 12-1 je příkladem vynášení sloupcového diagramu na jedné části horizontální osy a křivky na druhé části. Pro obě části horizontální osy jsou použita různá měřítka, zatímco vertikální osa slouží oběma grafům. To nám umožňuje vynášet části datové množiny, například, ve dvou odlišných formách. Obrázek 12-4 ukazuje výsledný výstup.

Prezentace dat v různých grafických formách může pomoci v interpretaci dat. Program 12-3 uvádí modulární návrh, který nám umožňuje volit typ grafu nebo diagramu, stejně tak jako vynášená data. Tento obecný návrh může být užitečný pro vytváření grafů různých forem se stejnými daty a experimentovat se změnami různých parametrů.



Obr. 12-3 Grafy mohou ukazovat současně pozitivní i negativní množství.

```

10  'PROGRAM 12-1. ROZTRZENY KOLACOVY DIAGRAM. KOLACOVY
20      'DIAGRAM S POPISEM A S ODTRZENOU LIBOVLONOU CASTI.
30  CLS
40  DIM N$(8), V(8)
50  PRINT "UVED MAXIMALNI HORIZONTALNI A VERTIKALNI"
60  PRINT "HODNOTY PRO TENTO ROZLISOVACI REZIM"
70  INPUT XM, YM
80  PRINT "UVED POSET ZNAKU, KTERE MOHOU TISKNUTY"
90  PRINT "PRES STINITKO OBRAZOVKY V TOMTO"
100 PRINT "ROZLISOVACIM REZIMU"
110 INPUT C1
120 PC = XM / C1  'PC JE POSET BODU NA ZNAK HORIZONTALNE
130 PRINT "UVED POSET MOZNYCH TISKOVYCH RADEK"
140 PRINT "MOZNYCH V TOMTO ROZLISOVACIM REZIMU"
150 INPUT R1
160 PR = YM / R1  'PR JE POSET BODU NA ZNAK VERTIKALNE
170 PRINT
180 PRINT "UVED SOURADNICE STREDU DIAGRAMU"
190 INPUT XN, YN
200 PRINT "UVED POLOMER"
210 INPUT R
220 IF XN+R > XM OR XN-R < 0 OR YN+R > YM OR YN-R < 0 THEN
    1020

```

```

230 PRINT
240      ***** VSTUP DAT *****
250 PRINT "UVED POCET ODDILO (MAX. 8)"
260 INPUT N
270 PRINT "UVED JMENO A HODNOTU KAZDEHO ODDILU"
280 T = 0
290      'VSTUP DAT. URCENI SOUCTU (T) VSECH HODNOT.
300 FOR K = 1 TO N
310     INPUT N$(K), V(K)
320     T = T + V(K)
330 NEXT K
340 PRINT "ODTRHNOUT KTERY ODDIL (1 -";N; ")"
350 INPUT E
360 IF E < 1 OR E > N THEN 340
370      ***** VYTVARENÍ KOLACOVÉHO DIAGRAMU *****
380 CLS
390 SCREEN 1
400 YA = 5 / 6   'YA JE KOREKCE ROZLISOVACI SCHOPNOSTI
410 B = 0         'B JE UHEL, KTERY URCUJE PREDCHOZI RADEK
420 S = 0
430 RE = 360 * 3.14159 / 180  'RE JE RADIANOVY EKVIVALENT
    360 STUPNU
440 FOR K = 1 TO N
450     XC = XN 'XN JE, YN JE STRED PRO NORMALNI CAST
460     YC = YN
470     'VYNASENA PRIMKA JE URCENA HODNOTOU TOHOTO
480     'ODDILU PLUS PREDCHAZEJICICH ODDILU
490     S = S + V(K)
500     'UHEL, KTERY SE MA POUZIT, JE URCEN
510     'PROCENTEM S / T KRUHU
520     A = RE * S / T
530     IF K <> E THEN 660  'MA BYT TENTO ODDIL ODTRZEN ?
540     AC = B + (A-B) / 2  'AC JE UHEL PRIMKY PULICI ODDIL
550     XE = XC + R/5 * COS(AC)  'ANO, TAK URCIT NOVY
        STRED (XE,YE) 1/5 VNE OD
560     YE = YC + R/5 * SIN(AC) * YA  'NORMALNIHO STREDU.
570     XP = XE + R * COS(B)  'PAK NAJIT KONCOVE BODY A
580     YP = YE + R * SIN(B) * YA  'NAKRESLIT.
590     LINE(XE,YE)-(XP,YP)
600     XP = XC + R * COS(A)
610     YP = YC + R * SIN(A) * YA

```

```

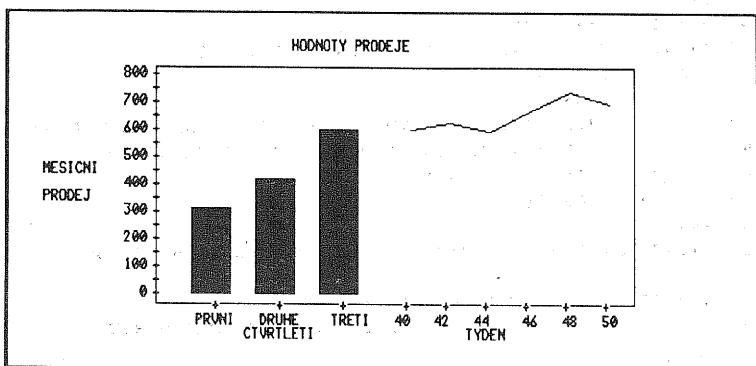
620      LINE(XC,YC)-(XP,YP)
630      XC = XE
640      YC = YE
650      '
660      FOR A1 = B TO A STEP 1/R
670          XP = XC + R * COS(A1)
680          YP = YC + R * SIN(A1) * YA
690          PSET(XP,YP)
700      NEXT A1
710      LINE(XC,YC)-(XP,YP)
720      'NATISKNOUT POPIS ODDILU
730          'NAJIT BOD 4 JEDNOTKY DALE SMEREM OD STREDU
740          'MIMO OBLOUK PATRICI K ODLIU
750          AC = B + (A - B) / 2   'AC JE UHEL OD HORIZONTALY
760          XL = XC + (R + 4) * COS(AC)   'KTERA BUDE PULIT
    ODDIL
770          YL = YC + (R + 4) * SIN(AC) * YA
780          '(XL,YL) JE BOD UZITY K UMISTENI POPISU
790          'UZIT TENTO BOD JAKO ZACATEK POPISU, JE-LI
800          'NA PRAVE STRANE KRUZNICE. JAKO KONEC POPISU,
810          'JE-LI VLEVO. JAKO STRED, JE-LI NAD NEBO POD
820          'KRUZNICI
830      IF XL > XC + 10 THEN 930
840          'BOD JE KONCEM POPISU
850      IF XL < XC - 10 THEN 900
860          'JINAK JE BOD STREDEM POPISU. KORIGUJ XL POMOCI
870          '1/2 POCTU BODU POTREBNYCH PRO POPIS
880          XL = XL - LEN(N$(K)) / 2 * PC
890      GOTO 930
900          'BOD JE KONCEM POPISU. PRESUNOUT ZPET O POSET
910          'BODU POTREBNYCH PRO POPIS
920          XL = XL - LEN(N$(K)) * PC
930          'PREVOD SOURADNIC V BODECH (XL,YL) NA CO NEJ-
940          'BLIZSI ODPOVIDAJICI TISKOVE POZICI
950          RO = INT(YL / PR)+1
960          CO = INT(XL / PC)+1
970          LOCATE RO,CO
980          PRINT N$(K);
990          B = A
1000 NEXT K
1010 GOTO 1030

```

```
1020 PRINT "SOURADNICE MIMO ROZSAH"
```

```
1030 END
```

Prog. 12-2 Kombinované grafy: sloupcový diagram a úsečkový graf.



Obr. 12-4 Kombinace sloupcového diagramu a úsečkového grafu, vytvořená programem Prog. 12-2, obsahuje jednu vertikální stupnice a dvě různé horizontální stupnice.

```
10 'PROGRAM 12-3. OBECNY GRAFICKY PROGRAM.  
20   'UMOZNUJE INTERAKTIVNI VSTUP DAT A VYBER TYPU  
30   'DIAGRAMU (USECKOVY, SLOUPCOVY, NEBO KRUHOVYCH  
40   'VYSECI). NAVRZEN PRO STINITKO 640 x 200 BODU.  
50   *****  
60 PC = 8    'PC JE POSET BODU NA ZNAK HOROZONTALNE  
70 PR = 8    'PR JE POSET BODU NA ZNAK VERTIKALNE  
80 YA = 5/12  'YA JE KOREKCE ROZLISOVACI SCHOPNOSTI  
90 CLS  
100 PRINT "1 - LINE      2 - BAR      3 - PIE"  
110 PRINT "JAKY DRUH DIAGRAMU";  
120 INPUT C$  
130 PRINT "NAZEV (TITUL) DIAGRAMU";  
140 INPUT T$  
150 SCREEN 2  
160 IF C$ = "1" OR C$ = "2" THEN GOSUB 210  
170 IF C$ = "1" THEN GOSUB 630  
180 IF C$ = "2" THEN GOSUB 740  
190 IF C$ = "3" THEN GOSUB 860
```

```

200 GOTO 1520
210 ##### USECKOVY NEBO SLOUPCOVY DIAGRAM #####
220 PRINT "UVED POSET ODDILU"
230 INPUT N
240 D = INT(586 / N)
250 PRINT "UVED JMENO A HODNOTU KAZDEHO ODDILU"
260 FOR K = 1 TO N
270   INPUT L$(K),S(K)
280   IF LEN(L$(K)) < D/8 THEN 320  'VEJDE SE TAM POPIS?
290   PRINT "POPIS PRILIS DLOUHY. MAX. DELKA JE";D/8
300   PRINT "NOVY POPIS";
310   INPUT L$(K)
320 NEXT K
330 PRINT "UVED MIN. A MAX. HODNOTY PRO VERTIKALNI OSU"
340 INPUT LO, HI
350 R = HI - LO  'R JE ROZSAH HODNOT
360 RS = (180 - 20) / R 'RS JE POMER UZITY PRO MERITKO
370 CLS
380 LINE(71,20)-(71,180)
390 LINE(71,180)-(639,180)
400 LINE(639,180)-(639,20)
410 LINE(639,20)-(71,20)
420 LOCATE 1,1  'TISK NAZVU DIAGRAMU (TITULU)
430 P = 40 - LEN(T$) / 2  'CENTROVANI NAZVU (TITULU)
440 PRINT TAB(P);T$
450 RO = 23
460 Y = 180
470 FOR K = 1 TO 5  'POPIS VERTIKALNI OSY SOUSEDNIMI
480   LOCATE RO,1  'PETINAMI Z ROZSAHU
490   L = LO + R * K / 5
500   PRINT USING "###.###";L
510   LINE(70,Y)-(639,Y)
520   RO = RO - 4
530   Y = Y - 32
540 NEXT K
550   'POPIS ODDILU
560 LOCATE 25,1
570 FOR K = 1 TO N
580   P = INT((72 + (K-1) * D + D/2) / PC - LEN(L$(K))/2
+ .5) + 1
590   PRINT TAB(P);L$(K);

```

```

600 NEXT K
610 RETURN
620 '
630 ##### USECKOVY DIAGRAM #####
640 FOR K = 1 TO N
650     Y(K) = INT((HI - S(K)) * RS + 20.5)
660 NEXT K
670 X = 71 + D/2
680 FOR K = 1 TO N-1
690     LINE(X,Y(K))-(X+D,Y(K+1))
700     X = X + D
710 NEXT K
720 RETURN
730 '
740 ##### SLOUPCOVY DIAGRAM #####
750 Y0 = INT(HI * RS + 20.5)      'ZJISTENI POLOHY 0
760 X1 = 71 + D/6
770 FOR K = 1 TO N
780     Y = INT((HI - S(K)) * RS + 20.5)
790     FOR X = X1 TO X1 + D * 2/3
800         LINE(X,Y)-(X,Y0)
810     NEXT X
820     X1 = X1 + D
830 NEXT K
840 RETURN
850 '
860 ##### KOLACOVY DIAGRAM #####
870 XC = 320
880 YC = 110
890 R = 160
900 PRINT "UVED POCET ODDILU (MAX. 6)"
910 INPUT N
920 PRINT "UVED JMENO A HODNOTU KAZDEHO ODDILU"
930 T = 0
940     'VSTUP DAT. ZJISTIT SOUCET (T) VSECH HODNOT.
950 FOR K = 1 TO N
960     INPUT NS(K), V(K)
970     IF V(K) > 0 THEN 1010
980     PRINT "HODNOTA MUSI BYT Kladna. UVED ZNOVA HODNOTU"
990     INPUT V(K)
1000 GOTO 970

```

```

1010      T = T + V(K)
1020  NEXT K
1030  CLS
1040  SCREEN 2
1050  LOCATE 1,1
1060  P = 40 - LEN(T$) / 2  'CENTROVANI NAZVU DIAGRAMU
1070  PRINT TAB(P);T$
1080  CIRCLE(XC,YC),R
1090  B = 0  'B JE UHEL, KTERY URCIL PREDCHAZEJICI USECKU
1100  S = 0
1110  RE = 360 * 3.14159 / 180  'RE JE EKVIVALENT 360
        STUPNU V RADIANECH
1120  FOR K = 1 TO N
1130      'USECKA, KTERA SE MA VYNEST, JE URCENA HODNO-
1140      'TOU TETO CASTI PLUS PREDCHAZEJICICH CASTI
1150  S = S + V(K)
1160      'UHEL, KTERY SE MA POUZIT JE PROCENTEM S / T
1170      'PLNEHO UHLU
1180  A = RE * S / T
1190  XP = XC + R * COS(A)
1200  YP = YC + R * SIN(A) * YA
1210  LINE(XC,YC)-(XP,YP)
1220      'UNISTENI POPISU K ODDILU
1230      'NALEZENI BODU 4 JEDNOTKY DALE OD STREDNHO
1240      'BODU OBLOUKU
1250  AC = B + (A - B) / 2  'AC JE UHEL OD HORIZONTALY
1260  XL = XC + (R + 4) * COS(AC)  'K OSSE UHLU ODDILU
1270  YL = YC + (R + 4) * SIN(AC) * YA
1280      '(XL,YL) JE BOD UZITY K UMISTENI POPISKU. UZIT
1290      'TENTO BOD JAKO ZACATEK POPISU, JE-LI NAPRAVO
1300      'OD KRUZNICE. JAKO KONCOVY BOD, JE-LI NALEVO.
1310      'JAKO STREDNI BOD, JE-LI NAD NEBO POD KRUZNICI
1320      'BOD JE ZACATKEM POPISU
1330  IF XL > XC + 10 THEN 1430
1340      'BOD JE KONCEM POPISU
1350  IF XL < XC - 10 THEN 1400
1360      'JINAK JE STREDEM POPISU. KORIGOVAT XL O 1/2
1370      'POCTU BODU NUTNYCH PRO POPIS
1380  XL = XL - LEN(N$(K)) / 2 * PC
1390  GOTO 1430
1400      'BOD JE KONCEM POPISU. PRESUNOUT ZPET O POCET

```

```

1410      'BODU NUTNYCH PRO POPIS
1420      XL = XL - LEN(N$(K)) * PC
1430      'PREVOD POLOHY V BODECH (XL,YL) NA NEJBLIZSI
1440      'ODPOVIDAJICI TISKOVOU POZICI
1450      RO = INT(YL / PR)+1
1460      CO = INT(XL / PC)+1
1470      LOCATE RO,CO
1480      PRINT N$(K);
1490      B = A
1500 NEXT K
1510 RETURN
1520 END

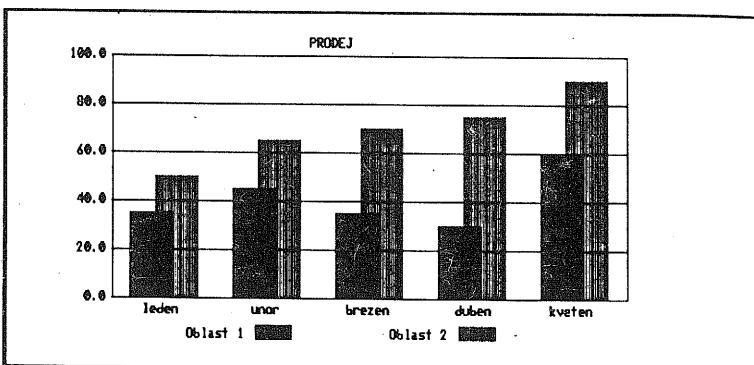
```

Prog. 12-3 Obecný grafický program - umožnuje volbu typu grafu.

## 10.2. Srovnávací grafy

Dvě, nebo více množin dat, můžeme vynášet do jednoho grafu pro srovnání vzájemných vztahů mezi proměnnými, jako jsou prodeje výrobků ve vztahu k obvodu nebo ve vztahu k prodavači. Existuje mnoho metod sestavení takových srovnávacích (komparativních) grafů.

Ve sloupcovém grafu můžeme srovnávat dvě množiny dat překrytím sloupců (Obr. 12-5). Programy pro vytváření tohoto typu srovnávacích grafů mohou vynášet každou datovou množinu jinou barvou, jako v Prog. 12-4. Sloupce pro jednu datovou množinu jsou poněkud posunuty a nakresleny přes sloupce druhé množiny.



Obr. 12-5 Srovnávací sloupcový graf z Prog. 12-4 ukazuje překrývající se sloupce, odpovídající dvěma množinám dat, vynesené ve stejných osách.

```

10  'PROGRAM 12-4. DVE MNOZINY DAT S PREKRYVAJICIMI SE
20    'SVISLYMI SLOUPCI. VYNASI SLOUPCE PRO DVE DATOVE
30    'MNOZINY, SDILEJICI STEJNE OSY X A Y. PREDPOKLADA
40    'VLIKOST STINITKA 640 x 200 BODU. DATOVE HODNOTY
50    'JSOU ZOBRAZENY DO BODU 12 - 172.
55    *****CLSS*****
60  CLS
70  DIM L$(8), S1(8), S2(8)
80  PC = 8   'PC JE POSET BODU NA ZNAK HORIZONTALNE
90    *****VSTUP POPISU A DAT DIAGRAMU *****
100 PRINT "NAZEV (TITUL) DIAGRAMU";
110 INPUT T$
120 PRINT "UVED. JMENO PRVE DATOVE MNOZINY"
130 INPUT T1$
140 PRINT "UVED JMENO DRUHE DATOVE MNOZINY"
150 INPUT T2$
160 PRINT "UVED POSET ODDILU"
170 INPUT N
180 D = INT(568 / N)  'D JE OBLAST PRO KAZDY ODDIL
190 PRINT "UVED POPIS A DVE HODNOTY PRO KAZDY ODDIL"
200 FOR K = 1 TO N
210   INPUT L$(K),S1(K),S2(K)
220   IF LEN(L$(K)) < D/PC THEN 260  'VEJDE SE TAM POPIS?

```

```

230 PRINT "POPIS JE PRILIS DLOUHY. MAX. DELKA JE";D/PC
240 PRINT "NOVOU HODNOTU";
250 INPUT L$(K)
260 NEXT K
270 PRINT "UVED MIN. A MAX. HODNOTY PRO VERTIKALNI OSU"
280 INPUT LO,HI
290 R = HI - LO   'R JE ROZSAH HODNOT
300 RS = (172 - 12) / R   'RS JE POMER PRO MERTKO
310 '***** KRESLENI SOURADNE SITE A POPISU *****
320 CLS
330 SCREEN 2
340 LINE(71,12)-(71,172)
350 LINE(71,172)-(639,172)
360 LINE(639,172)-(639,12)
370 LINE(639,12)-(71,12)
380 LOCATE 1,1 'TISK NAZVU DIAGRAMU (TITULU)
390 P = 40 - LEN(T$) / 2 'CENTROVANI TITULU
400 PRINT TAB(P);T$
410 RO = 22
420 Y = 172
430 FOR K = 0 TO 5   'POPSAI VERTIKALNI OSY SOUSEDNIMI
440   LOCATE RO,3   'PETINAMI ROZSAHU
450   L = LO + R * K / 5
460   PRINT USING "####.#";L
470   LINE(70,Y)-(639,Y)
480   RO = RO - 4
490   Y = Y - 32
500 NEXT K
510   'POPIS ODDILU
520 LOCATE 23,1
530 FOR K = 1 TO N
540   P = INT((71+(K-1)*D+D/2)/PC - LEN(L$(K))/2 + .5) + 1
550   PRINT TAB(P);L$(K);
560 NEXT K
570   'VYTVARENI KODOVYCH OBLASTI
580 LOCATE 25,28-LEN(T1$)
590 PRINT T1$;
600 REM
610 FOR X = 228 TO 268
620   LINE(X,190)-(X,199),7
630 NEXT X

```

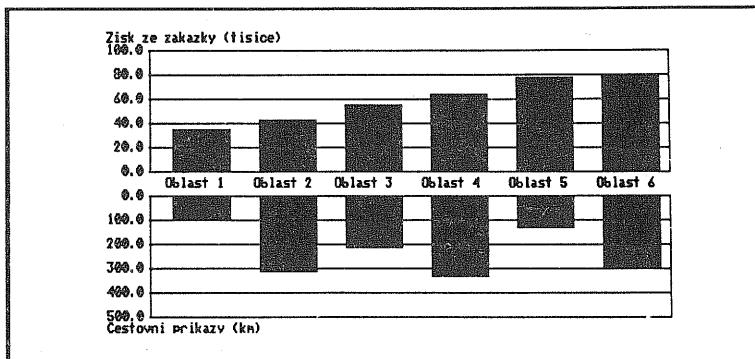
```

640 LOCATE 25,56-LEN(T2$)
650 PRINT T2$;
660 REM
670 FOR X = 452 TO 492 STEP 2
680 LINE(X,190)-(X,199)
690 NEXT X
700 ***** VYTVORENI SLOUPCU *****
710 Y0 = INT(HI * RS + 12.5) 'NALEZENI KDE JE 0
720 X1 = 71 + D/6
730 FOR K = 1 TO N
740 Y = INT((HI - S1(K)) * RS + 12.5)
750 REM
760 FOR X = X1 TO X1 + D * 5/12
770 LINE(X,Y)-(X,Y0),7
780 NEXT X
790 X2 = X1 + D * 3/12
800 Y = INT((HI - S2(K)) * RS + 12.5)
810 REM
820 FOR X = X2 TO X2 + D * 5/12 STEP 2
830 LINE(X,Y)-(X,Y0)
840 NEXT X
850 X1 = X1 + D
860 NEXT K
870 END

```

Prog. 12-4 Srovnávací graf: překryté sloupcové diagramy.

Jiný typ srovnávacího (komparativního) sloupcového diagramu je uveden na Obr. 12-6. Tento graf byl vytvořen pomocí Prog. 12-5. V uvedeném příkladu sdílí dvě množiny dat společnou osu X, ale mají různé osy Y. Toto uspořádání nám umožňuje použít různá měřítka pro každou množinu dat.



Obr. 12-6 Srovnávací graf, zobrazený programem Prog 12-5, se sloupci kreslenými pro jednu množinu dat směrem nahoru a dolů pro druhou

```

10 'PROGRAM 12-5. SROVNAVACI SLOUPCOVY GRAF SE DVEMA STUP-
20 'NICEMI Y. VKLADAJI SE NAZVY A MNOZINY DAT PRO
30 'LIBOVOLNY POSET ODDILU. KRESLI SLOUPCE PRO JEDNU
40 'MNOZINU DAT VERTIKALNE V HORNÍ POLOVINE STINITKA
50 'A SLOUPCE PRO DRUHOU MNOZINU DAT VERTIKALNE V
60 'DOLNI POLOVINE. NAVRZENO PRO STINITKO 640 x 200
70 'BODU.
80 ****
90 CLS
100 DIM L$(15), T(15), B(15)
110 PC = 8 'PC JE POSET BODU NA ZNAK HORIZONTALNE
120 ***** VSTUP DAT *****
130 PRINT "UVED POSET ODDILU"
140 INPUT N
150 D = INT(568 / N) 'D JE POVOLENY POSET BODU NA ODDIL
160 PRINT "UVED JMENO A HODNOTY KAZDEHO ODDILU"
170 FOR K = 1 TO N
180 INPUT L$(K), T(K), B(K)
190 IF LEN(L$(K)) < D/PC THEN 230 'VELDE SE TAM POPIS?
200 PRINT "POPIS PRILIS DLOUHY. MAX. DELKA JE"; D/PC
210 PRINT "NOVY POPIS";
220 INPUT L$(K)
230 NEXT K

```

```

240 PRINT "UVED POPIS PRO HORNI VERTIKALNI OSU"
250 INPUT TS
260 PRINT "UVED MIN. A MAX. HODNOTY PRO HORNI VERT. OSU"
270 INPUT LT, HT
280 R1 = HT - LT
290 RT = (92 - 12) / R1 'RT JE POMER-MERITKO HORNI CASTI
300 PRINT "UVED POPIS PRO DOLNI VERTIKALNI OSU"
310 INPUT BS
320 PRINT "UVED MIN. A MAX. HODNOTY PRO DOLNI VERT. OSU"
330 INPUT LB, HB
340 R2 = HB - LB
350 RB = (188 - 108) / R2 'RB JE POMER-MERITKO DOLNI CASTI
360 '***** KRESLENI POZADI *****
370 CLS
380 SCREEN 2
390 LINE(71,12)-(71,92)
400 LINE(71,92)-(639,92)
410 LINE(639,92)-(639,12)
420 LINE(639,12)-(71,12)
430 LINE(71,108)-(71,188)
440 LINE(71,188)-(639,188)
450 LINE(639,188)-(639,108)
460 LINE(639,108)-(71,108)
470 LOCATE 1,4 'TISK HORNIHO POPISU
480 PRINT TS
490 RO = 12
500 Y = 92
510 FOR K = 0 TO 5 'POPIS VERTIKALNI OSY SOUSEDNIMI
520 LOCATE RO,3 'PETINAMI ROZSAHU
530 L = LT + R1 * K / 5
540 PRINT USING "####.#";L;
550 LINE(70,Y)-(639,Y)
560 RO = RO - 2
570 Y = Y - 16
580 NEXT K
590 LOCATE 25,4 'POPIS DOLNI POLOVINY
600 PRINT BS;
610 RO = 14
620 Y = 108
630 FOR K = 0 TO 5
640 LOCATE RO,3

```

```

650      L = LB + R2 * K / 5
660      PRINT USING "####.#";L;
670      LINE(70,Y)-(639,Y)
680      RO = RO + 2
690      Y = Y + 16
700 NEXT K
710  'POPIS ODDILU
720 LOCATE 13,1
730 FOR K = 1 TO N
740      P = INT((72 + (K-1) * D + D/2) / PC -
LEN(L$(K))/2 + .5) + 1
750      PRINT TAB(P);L$(K);
760 NEXT K
770 ***** KRESLENI SLOUPCU *****
780 X1 = 96
790 FOR K = 1 TO N
800      YT = INT((HT - T(K)) * RT + 12.5)
810      YB = INT(B(K) * RB + 108.5)
820      FOR X = X1 TO X1 + D * 2/3
830          LINE(X,YT)-(X,92)
840          LINE(X,YB)-(X,108)
850      NEXT X
860      X1 = X1 + D
870 NEXT K
880 END

```

Prog. 12-5 Srovnávací graf ze dvou sloupcových grafů: jeden nahoru a druhý dolů.

Křivky mohou být srovnávány jednoduchým vynesením několika množin dat do téhož grafu. Také můžeme konstruovat kumulativní plošné diagramy metodami demonstrovanými v Prog. 12-6. Výsledný graf, uvedený na Obr. 12-7, obsahuje horní křivku jako součet dvou množin dat a užívá stínování pro rozlišení těchto oblastí. Tento typ grafu může často vést k chybné interpretaci, protože horní čára ukazuje součet obou množin dat a nikoli skutečné hodnoty dat.

```

10  'PROGRAM 12-6. PLOSNY DIAGRAM PRO DVE MNOZINY KUMULATIVNICH DAT. VYNASI HORNI KRIVKU JAKO SOUCET
20

```

```

30      'PRVE A DRUHE DATOVE MNOZINY. PREDPOKLADANA VELI-
40      'KOST STINITKA JE 640 x 200 BODU.
50      ****
60  CLS
70  DIM L$(20), S1(20), S2(20)
80  PC = 8   'PC JE POSET BODU NA ZNAK HORIZONTALNE
90      ***** VSTUP DAT A POPISEK ****
100 PRINT "NAZEV DIAGRAMU";
110 INPUT TS
120 PRINT "UVED JMENO PRVE MNOZINY DAT"
130 INPUT T1$
140 PRINT "UVED JMENO DRUHE MNOZINY DAT"
150 INPUT T2$
160 PRINT "UVED POSET HORIZONTALNICH ODDILU"
170 INPUT N
180 D = INT(568 / N)
190 PRINT D
200 PRINT "UVED POPIS A DVE HODNOTY PRO KAZDY ODDIL"
210 M2 = 0   'M2 JE MAX. HODNOTA V POLI S2
220 FOR K = 1 TO N
230     INPUT L$(K), S1(K), S2(K)
240     S2(K) = S1(K) + S2(K)
250     IF S2(K) <= M2 THEN 270
260     M2 = S2(K)
270     IF LEN(L$(K)) < D/PC THEN 310  'VEJDE SE TAM POPIS?
280     PRINT "POPIS PRILIS DLOUHY. MAX. DELKA JE"; D/PC
290     PRINT "NOVY POPIS";
300     INPUT L$(K)
310 NEXT K
320 PRINT "UVED MIN. A MAX. HODNOTY PRO VERTIKALNI OSU"
330 INPUT LO, HI
340 IF HI >= M2 THEN 380  "JE HI DOSTI VELKE?
350 PRINT "MAX. HODNOTA NENI DOST VELKA. UVED ZNOVA"
360 INPUT HI
370 GOTO 340
380 R = HI - LO  'R JE ROZSAH HODNOT
390 RS = (172 - 12) / R  'RS JE POMER UZITY PRO MERITKO
400      ***** KRESLENI SOURADNE SITE A POPISU *****
410 CLS
420 SCREEN 2
430 LINE(71,12)-(71,172)

```

```

440 LINE(71,172)-(639,172)
450 LINE(639,172)-(639,12)
460 LINE(639,12)-(71,12)
470 LOCATE 1,1      'TISK TITULU
480 P = 40 - LEN(T$) / 2    'CENTROVANI TITULU
490 PRINT TAB(P);T$
500 RO = 22
510 Y = 172
520 FOR K = 0 TO 5    'POPSAT VERTIKALNI OSU SOUSEDNIMI
530     LOCATE RO,3   'HODNOTAMI ZOBRAZOVANEHO ROZSAHU
540     L = LO + R * K / 5
550     PRINT USING "####.#";L
560     LINE(70,Y)-(639,Y)
570     RO = RO - 4
580     Y = Y - 32
590 NEXT K
600     'POPIS VSECH ODDILU
610 LOCATE 23,1
620 FOR K = 1 TO N
630     P = INT((71+(K-1)*D+D/2)/PC - LEN(L$(K))/2 + .5)+1
640     PRINT TAB(P);L$(K);
650 NEXT K
660     'VYTVARENI KODOVYCH OBLASTI
670 LOCATE 25,28-LEN(T1$)
690 PRINT T1$;
700 FOR X = 228 TO 288
710     LINE(X,190)-(X,199)
720 NEXT X
730 LINE(228,190)-(288,190),2
740 LINE(228,199)-(288,199),2
750 LOCATE 25,56-LEN(T2$)
760 REM
770 PRINT T2$;
780 FOR X = 452 TO 512 STEP 2
790     LINE(X,190)-(X,199)
800 NEXT X
810 LINE(452,190)-(512,190),5
820 LINE(452,199)-(512,199),5
830     ***** VYTVARENI SLOUPCU *****
840 Y0 = 171
850 X1 = 71 + D/2    'POCAT. BOD DO 1/2 PRVEHO ODDILU

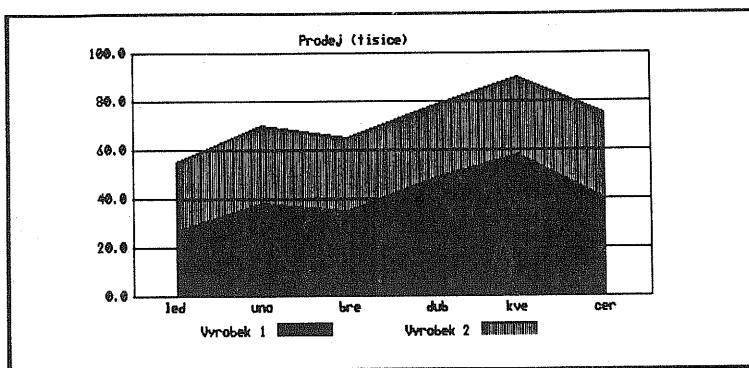
```

```

860 Y1 = INT((HI - S2(1)) * RS + 12.5)
870 REM
880 FOR K = 2 TO N
890     X2 = X1 + D
900     Y2 = INT((HI - S2(K)) * RS + 12.5)
910     M = (Y2-Y1) / (X2-X1)
920     B = Y1 + 1 - M * X1
930     FOR X = X1 TO X2 STEP 2
940         Y = M * X + B
945         LINE (X,Y)-(X,Y0)
950     NEXT X
960     LINE(X1,Y1)-(X2,Y2)      'NAKRESLIT KRIVKU
970     X1 = X2                  'ULOZIT OKAMZITU HODNOTU
980     Y1 = Y2                  'SOURADNIC BODU
990 NEXT K
1000 'KRESLENI DOLNI KRIVKY (PRVA MNOZINA DAT)
1010 X1 = 71 + D/2
1020 Y1 = INT((HI - S1(1)) * RS + 12.5)
1030 REM
1040 FOR K = 2 TO N
1050     X2 = X1 + D
1060     Y2 = INT((HI - S1(K)) * RS + 12.5)
1070     M = (Y2 - Y1) / (X2 - X1)
1080     B = Y1 + 1 - M * X1
1090     FOR X = X1 TO X2
1100         Y = M * X + B
1110         LINE(X,Y)-(X,Y0)
1120     NEXT X
1130     LINE(X1,Y1)-(X2,Y2)      'KRESLENI KRIVKY
1140     X1 = X2
1150     Y1 = Y2
1160 NEXT K
1170 END

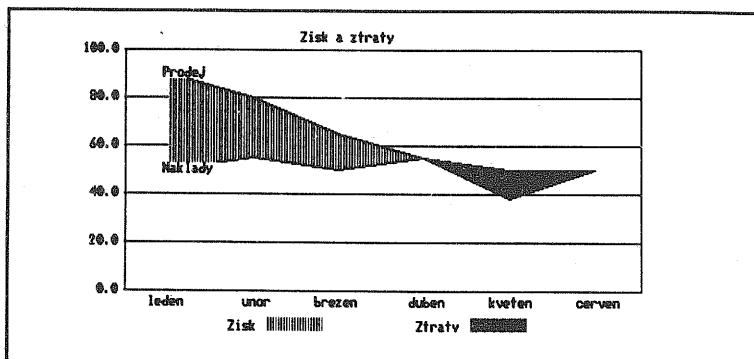
```

Prog. 12-6 Kumulativní plošný diagram, vynášející dvě množiny dat.



Obr. 12-7 Kumulativní plošný diagram vytořený programem Prog. 12-6. Dolní křivka odpovídá prvé datové množině a horní, součtu hodnot obou množin.

Stínovací vzory využíváme ke zvýraznění oblastí mezi křivkami tak, jak je uvedeno na Obr. 12-8. Tento pásový diagram byl vytvořen programem Prog. 12-7. Tuto techniku můžeme použít na pomoc při odhadování rozdílů. V tomto příkladu jsou použity dva různé stínovací vzory k rozlišení zisků od ztrát (tj., klesne-li jedna křivka pod druhou).



Obr. 12-8 Pásový diagram vynesený programem Prog. 12-7 ilustruje užití stínování mezi křivkami.

```

10  'PROGRAM 12-7. STINOVANY PASOVY DIAGRAM PRO DVE MNOZINY
20  'DAT. VYNASI KRIVKY PODLE DVOU DATOVYCH MNOZIN.
30  'STINUJE OBLASTI PRESAHU KRIVEK. PREDP. VELIKOST
35  'STINITKA JE 640 X 200 BODU.
40  ****
50  CLS
60  DIM L$(20), S1(20), S2(20)
70  PC = 8   'PC JE POSET BODU NA ZNAK HORIZONTALNE
80  ***** VSTUP DAT A POPISEK *****
90  PRINT "NAZEV DIAGRAMU";
100 INPUT T$
110 PRINT "UVED JMENO PRVE MNOZINY DAT"
120 INPUT T1$
130 PRINT "UVED JMENO DRUHE MNOZINY DAT"
140 INPUT T2$
150 PRINT "UVED JMENO OBLASTI, KDE JSOU PRVA DATA VETSI
      NEZ DRUHA"
160 INPUT T3$
170 PRINT "UVED JMENO OBLASTI, KDE JSOU DRUHA DATA VETSI
      NEZ PRVA"
180 INPUT T4$
190 PRINT "UVED POSET HORIZONTALNICH ODDILU"
200 INPUT N

```

```

210 D = INT(568 / N)           'D JE POSET BODU NA ODDIL
220 IF D/2 <> INT(D/2) THEN D=INT(D/2+.5)*2   'D MA BYT SUDE
230 PRINT "UVED POPIS A DVE HODNOTY PRO KAZDY ODDIL"
240 FOR K = 1 TO N
250     INPUT L$(K),S1(K),S2(K)
260     IF LEN(L$(K)) < D/PC THEN 300  'VEJDE SE TAM POPIS?
270     PRINT "POPIS PRILIS DLOUHY. MAX. DELKA JE";D/PC
280     PRINT "NOVY POPIS";
290     INP
300 NEXT K
310 PRINT "UVED MIN. A MAX. HODNOTY PRO VERTIKALNI OSU"
320 INPUT LO, HI
330 R = HI - LO                'R JE ROZSAH HODNOT
340 RS = (172 - 12) / R         'RS JE POMER UZITY PRO MERITKO
350     '***** KRESLENI SOURADNE SITE A POPISU *****
360 CLS
370 SCREEN 2
380 LINE(71,12)-(71,172)
390 LINE(71,172)-(639,172)
400 LINE(639,172)-(639,12)
410 LINE(639,12)-(71,12)
420 LOCATE 1,1 'TISK TITULU
430 P = 40 - LEN(T$) / 2        'CENTROVANI TITULU
440 PRINT TAB(P);T$
450 RO = 22
460 Y = 172
470 FOR K = 0 TO 5             'POPSAT VERTIKALNI OSU SOUSEDNIMI
480     LOCATE RO,3            'HODNOTAMI ZOBRAZOVANEHO ROZSAHU
490     L = LO + R * K / 5
500     PRINT USING "####.#";L
510     LINE(70,Y)-(639,Y)
520     RO = RO - 4
530     Y = Y - 32
540 NEXT K
550     'POPIS VSECH ODDILU
560 LOCATE 23,1
570 FOR K = 1 TO N
580     P = INT((71+(K-1)*D+D/2)/PC - LEN(L$(K))/2 + .5)+1
590     PRINT TAB(P);L$(K);
600 NEXT K
610     'VYTVARENIA KODOVYCH OBLASTI

```

```

620 LOCATE 25,28-LEN(T3$)
630 PRINT T3$;
640 FOR X = 228 TO 288 STEP 3
650     LINE(X,190)-(X,199)
660 NEXT X
670 LOCATE 25,56-LEN(T4$)
680 PRINT T4$;
690 FOR X = 452 TO 512 STEP 2
700     LINE(X,190)-(X,199)
710 NEXT X
720     ***** VYTVARENI SLOUPCU *****
730 Y0 = 171
740 X1 = 71 + D/2      'POCAT. BOD DO 1/2 PRVEHO ODDILU
750 Y1 = INT((HI - S1(1)) * RS + 12.5)
760 R1 = INT(Y1/PC + .5)
770 C1 = INT(X1/PC + .5)  'X1,Y1 A X2,Y2 PATRI PRVE KRIVCE.
780 X3 = X1            'X3,Y3 A X4,Y4 PATRI CRUHE KRIVCE
790 Y3 = INT((HI - S2(1)) * RS + 12.5)
800 R3 = INT(Y3/PC + .5)
810 C3 = INT(X3/PC + .5)
820 FOR K = 2 TO N
830     X2 = X1 + D
840     Y2 = INT((HI - S1(K)) * RS + 12.5)
850     M1 = (Y2-Y1) / (X2-X1)          'SMERNICE A PRUSECIK
860     B1 = Y1 - M1 * X1             'PRVE KRIVKY
870     X4 = X2
880     Y4 = INT((HI - S2(K)) * RS + 12.5)
890     M3 = (Y4-Y3) / (X4-X3)          'SMERNICE A PRUSECIK
900     B3 = Y3 - M3 * X3             'DRUHE KRIVKY
910     'STINOVACI VZOR (KAZDA DRUHA NEBO TRETI USECKA) JE
920     'URCEN KRIVKOU, KTERA JE NAHORE. KRESLIT KAZDOU
930     'DRUHOU USECKU, JE-LI KRIVKA 3-4 NAHORE A KAZDOU
935     'TRETI, JE-LI KRIVKA 1-2 NAHORE.
940     IF Y3 <= Y1 AND Y4 <= Y2 THEN 1050 'KRIVKA 3-4
         VYSE NEZ 1-2
950     IF Y3 > Y1 AND Y4 > Y2 THEN 1070 'KRIVKA 1-2 VYSE
         NEZ 3-4
960     'JINAK SE OBE KRIVKY KRIZUJI
970     XP = INT((B1-B3) / (M3-M1) + .5) 'URCENI X
         PRUSECIKU KRIVEK
980     IF Y3 > Y1 THEN 1020

```

```

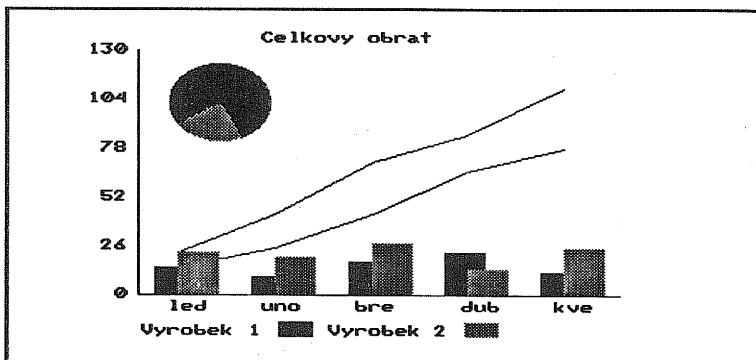
990      B = X3: E = XP: I = 2: GOSUB 1230 'KРИVKA 3-4 JE
        NAHORE DO XP
1000     B = XP: E = X4: I = 3: GOSUB 1230 'KРИVKA 1-2 JE
        NAHORE OD XP
1010     GOTO 1080
1020     B = X3: E = XP: I = 3: GOSUB 1230
1030     B = XP: E = X4: I = 2: GOSUB 1230
1040     GOTO 1080
1050     B = X3: E = X4: I = 2: GOSUB 1230
1060     GOTO 1080
1070     B = X3: E = X4: I = 3: GOSUB 1230
1080     LINE(X1,Y1)-(X2,Y2)           'NAKRESLIT OBE KРИVKY
1090     LINE(X3,Y3)-(X4,Y4)
1100     X1 = X2
1110     Y1 = Y2
1120     X3 = X4
1130     Y3 = Y4
1140 NEXT K
1150 LOCATE R1,C1                  'POPIS ZACASKU KРИVEK
1160 PRINT T1$;
1170 LOCATE R3,C3
1180 PRINT T2$;
1190 GOTO 1330
1200 '
1210 ##### STINOVANI OBLASTI #####
1220 REM
1230 REM                               'VYMAZANI CAR
        SOURADNE SITE
1235 FOR X = B TO E
1240     YF = M1 * X + B1
1250     YS = M3 * X + B3
1260     LINE(X,YF)-(X,YS),0
1270     IF X/I <> INT(X/I) THEN 1300 'JE CARA NASOBKEM I ?
1280 REM
1290     LINE(X,YF)-(X,YS),1
1300 NEXT X
1310 RETURN
1320 #####
1330 END

```

Prog. 12-7 Stínaný pásový diagram pro dvě množiny.

### 10.3. Vícenásobné formáty

Technikou, užitečnou pro srovnávání dvou množin dat, je vynesení těchto dat v několika různých formátech do téhož grafu. Tato metoda umožnuje zobrazovat různé typy vzájemných vztahů mezi datovými množinami. Kombinace sloupcového diagramu, úsečkového grafu a diagramu kruhových výsečí, uvedená na Obr. 12-9, byla vytvořena programem Prog. 12-8. Relativní velikost mezi těmito dvěma datovými množinami je zobrazena sloupcovým diagramem. Celkové součty jsou zobrazeny úsečkovým grafem. Diagram kruhových výsečí ukazuje celkový procentuální podíl obou množin. Pro identifikaci obou množin dat je užito barevného kódu.



Obr. 12-9 Graf vícenásobného formátu, vytvořený programem Prog. 12-8, poskytuje několik typů srovnání.

```
10  'PROGRAM 12-8. DVE DATOVE MNOZINY VE VICENASOBNYCH FOR-
20  'MATECH. VSTUPUJUJI DVE DATOVE MNOZINY. VYNASI DATA
30  'VE FORME SLOUPCOVEHO DIAGRAMU, KOLACOVEHO DIAGRA-
40  'MU UKAZUJUJICIHO PROCENTA CELKU A VE TVARU KRIVKY
50  'PRO KUMULATIVNI HODNOTY V CASE. PREDPOKLADANA
```

```

60      'VELIKOST STINITKA 320 X 200 BODU.
65      ****
70  CLS
80  DIM L$(5), S1(5), S2(5), C1(5), C2(5)
90  PC = 8      'PC JE POSET BODU NA ZNAK HORIZONTALNE
100 XC = 73     'XC,YC JE STRED KOLACOVEHO DIAGRAMU
110 YC = 40
120 RA = 30     'RA JE POLOMER KOLACOVEHO DIAGRAMU
130 YA = 5/6    'YA JE KOREKCE ROZLIS. SCHOPNOSTI NUTNA PRO Y
140      ***** VSTUP DAT A POPISEK *****
150 PRINT "NAZEV DIAGRAMU";
160 INPUT T$
170 PRINT "UVED JMENO PRVE MNOZINY DAT"
180 INPUT T1$
190 PRINT "UVED JMENO DRUHE MNOZINY DAT"
200 INPUT T2$
210 PRINT "UVED POSET ODDILU"
220 INPUT N
230 D = INT(290 / N)      'D JE POSET BODU NA ODDIL
240 T1 = 0                  'T1 A T2 JSOU UZITY PRO SCITANI
250 T2 = 0
260 PRINT "UVED POPIS A DVE HODNOTY PRO KAZDY SLOUPEC"
270 FOR K = 1 TO N
280   INPUT L$(K),S1(K),S2(K)
290   T1 = T1 + S1(K)
300   C1(K) = T1
310   T2 = T2 + S2(K)
320   C2(K) = T2
330   IF LEN(L$(K)) < D/PC THEN 370  'VEJDE SE TAM POPIS?
340   PRINT "POPIS PRILIS DLOUHY. MAX. DELKA JE";D/PC
350   PRINT "NOVY POPIS";
360   INPUT L$(K)
370 NEXT K
380 PRINT "UVED MIN. A MAX. HODNOTY PRO VERTIKALNI OSU"
390 INPUT LO, HI
400 IF HI >= T1 AND HI >= T2 THEN 440
410 PRINT "MAXIMUM NENI DOST VELKE. UVED ZNOVA MAXIMUM"
420 INPUT HI
430 GOTO 400
440 R = HI - LO          'R JE ROZSAH HODNOT
450 RS = (172 - 12) / R    'RS JE POMER UZITY PRO MERITKO

```

```

460 ***** KRESLENI SOURADNE SITE A POPISU *****
470 CLS
480 SCREEN 1
490 LINE(30,12)-(30,172)
500 LINE(30,172)-(319,172)
510 LOCATE 1,1 'TISK TITULU
520 P = 20 - LEN(T$) / 2 'CENTROVANI TITULU
530 PRINT TAB(P);T$
540 RO = 22
550 Y = 172
560 FOR K = 0 TO 5      'POPSAT VERTIKALNI OSU SOUSEDNIMI
570     LOCATE RO,1      'HODNOTAMI ZOBRAZOVANEHO ROZSAHU
580     L = LO + R * K / 5
590     PRINT USING "###";L
600     RO = RO - 4
610     Y = Y - 32
620 NEXT K
630     'POPIS VSECH ODDILU
640 LOCATE 23,1
650 FOR K = 1 TO N
660     P = INT((30+(K-1)*D+D/2)/PC - LEN(L$(K))/2 + .5)+1
670     PRINT TAB(P);L$(K);
680 NEXT K
690     'VYTVARENI KODOVYCH OBLASTI
700 LOCATE 25,14-LEN(T1$)
710 REM          'P
720 PRINT T1$;
730 FOR X = 114 TO 134
740     LINE(X,190)-(X,199),2
750 NEXT X
760 LOCATE 25,28-LEN(T2$)
770 REM          'DRUHA MNOZINA DAT SE KRESLI ZELENE
780 PRINT T2$;
790 FOR X = 226 TO 246
800     LINE(X,190)-(X,199),6
810 NEXT X
820
830     ***** VYTVARENI SLOUPCU *****
840 Y0 = 171
850 X1 = 30 + D/6
860 FOR K = 1 TO N

```

```

870      Y = INT((HI - S1(K)) * RS + 12.5)
880      REM
890      FOR X = X1 TO X1 + D * 5/12
900          LINE(X,Y)-(X,Y0),2
910      NEXT X
920      X2 = X1 + D * 3/12      'DRUHY SLOUPEC ZACINA 3/12 OD
PRVEHO
930      Y = INT((HI - S2(K)) * RS + 12.5)
940      REM
950      FOR X = X2 TO X2 + D * 5/12
960          LINE(X,Y)-(X,Y0),4
970      NEXT X
980      X1 = X1 + D
990 NEXT K
1000 '
1010      ***** VYTWARENI USECEK *****
1020 X = 30 + D * 5 / 12
1030 Y1 = INT((HI - C1(1)) * RS + 12.5)
1040 Y2 = INT((HI - C2(1)) * RS + 12.5)
1050 FOR K = 2 TO N
1060      Y3 = INT((HI - C1(K)) * RS + 12.5)
1070      Y4 = INT((HI - C2(K)) * RS + 12.5)
1080      REM
1090      LINE(X,Y1)-(X+D,Y3),2
1100      REM
1110      LINE(X,Y2)-(X+D,Y4),4
1120      X = X + D
1130      Y1 = Y3
1140      Y2 = Y4
1150 NEXT K
1160 '
1170      ***** VYTVORENI KOLACOVEHO DIAGRAMU *****
1180 CT = T1 + T2           'CT JE SOUCET OBOU MNOZIN
1190      'A1 JE PROCENTO KRUHU (V RADIANECH) ODPOVIDAJICI T1
1200 A1 = 360 * 3.14159 / 180 * T1 / CT
1210 DA = 1/RA/15          'KROK MUSI BYT VELMI MALY, ABY
      SE KRUZNICE UPLNE VYPLNILA
1220 REM
1230 FOR A = DA TO A1 STEP DA
1240      X = XC + RA * COS(A)
1250      Y = YC + RA * SIN(A) * YA

```

```

1260      LINE(XC,YC)-(X,Y),2
1270 NEXT A
1280   'VYPLNIT ZBYTEK KRUZNICE JINOU BARVOU (PRO T2)
1290 REM
1300 FOR A = A1+DA TO 6.28318 STEP DA
1310   X = XC + RA * COS(A)
1320   Y = YC + RA * SIN(A) * YA
1330   LINE(XC,YC)-(X,Y),4
1340 NEXT A
1350 END

```

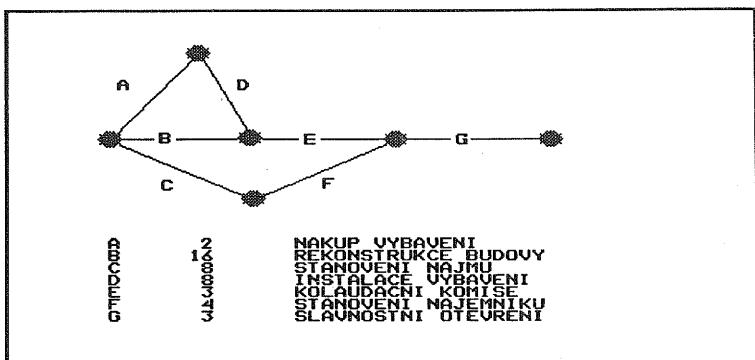
Prog. 12-8 Vícenásobné formáty: sloupcový diagram s překrytím, součtový úsečkový graf a diagram kruhových výsečí.

#### **10.4. Grafy řízení projektů**

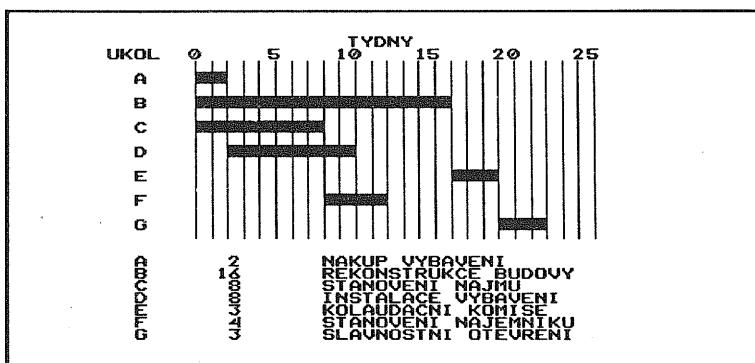
---

Grafy, zobrazující síť úloh projektu, jako na Obr. 12-10, mohou být užity na pomoc při plánování a kontrole úloh. Tento typ grafu je zvláště užitečný při užití plánovacích metod projektů. Úlohy jsou v grafu sítě reprezentovány hranami a jsou uspořádány zleva doprava podle toho, kdy mohou být zahájeny. Kružnice (uzly) jsou použity pro označení zahájení a ukončení úloh. Kružnice nejvíce vlevo ukazuje začátek projektu, kdy mohou být současně zahájeny úlohy A, B a C. Úloha D nemůže být zahájena dříve, dokud není dokončena úloha A a úloha E musí čekat na úlohy A, B a D.

Úlohy projektu také mohou být znázorněny v časovém diagramu, který ukazuje skutečná data zahájení a ukončení. Diagram projekčního času, uvedený na Obr. 12-11, byl vytvořen programem Prog. 12-9. Tento diagram ukazuje relativní začátky a konce týdnů pro každou úlohu projektu. Úlohy jsou reprezentovány horizontálními sloupci o délce úměrné plánovanému času úlohy. Oba uvedené diagramy jsou užitečné pro plánování a řízení projektů.



Obr. 12-10 Síťový graf, ukazující posloupnost různých úloh v projektu.



Obr. 12-11 Časový diagram, vytvořený Prog. 12-9, ukazuje data zahájení a ukončení plánovaných úloh projektu.

```

10  'PROGRAM 12-9. CASOVY DIAGRAM.
20      'PREDPOKLADANA VELIKOST STINITKA 320 x 200 BODU
30      'SE 40 ZNAKY NA RADEK. CASOVE SLOUPCE JSOU ZOBR-
40      'ZENY DO BODU 60 - 300.
50      '*****

```

```

60  CLS
70  SCREEN 1
80  READ TDS          'POPIS CASOVEHO ODDILU
90  READ ET,LT         'POPISY 1. A POSLEDNIHO CAS. ODDILU
100 RT = LT - ET      'RT JE ROZSAH CASU
110 RS = (300 - 60) / RT  'RS JE POMER UZITY PRO MERITKO
120 LOCATE 1,20
130 PRINT TDS;
140 LOCATE 2,2
150 PRINT "UKOL";
160 CO = 6
170 X = 60
180 FOR K = 0 TO 5    'POPSAT SLOUPCE PO 1/5 ROZSAHU CASU
190     LOCATE 2,CO
200     T = ET + RT * K/5
210     PRINT USING "###";T;
220     X = X + 48
230     CO = CO + 6
240 NEXT K
250 I = 240 / 25      'VYTvorit 25 USECEK NA 240 BODU
260 X = 60
270 FOR L = 1 TO 26   '26 RADEK CELKEM
280     LINE(X,16)-(X,132)
290     X = X + I
300 NEXT L
310 R1 = 19           'R1 JE RADA PRO TISK TABULKY (SPODEK
                      STINITKA)
320 R2 = 4            'R2 JE RADA PRO POPISY SLOUPCU
                      (VRSEK STINITKA)
330 FOR K = 1 TO 7
340     READ C$,T$,S,D  'KOD, JMENO ULOHY, ZACATEK, DELKA
350     LOCATE R1,4      'VYTVORENI TABULKY NA SPODKU STIN.
360     PRINT C$;
370     PRINT TAB(10); USING "##";D;
380     PRINT TAB(18);T$;
390     LOCATE R2,4      'VYTVORENI SLOUPCU NAHORE
400     PRINT C$;
410     X1 = S * RS + 60 'URCENI POLOHY ZACATKU SLOUPCE
420     X2 = (S + D) * RS + 60 'URCENI POLOHY KONCE
430     Y1 = (R2 - 1) * 8 - 1  'HODNOTY Y JSOU ZALOZENY
440     Y2 = Y1 + 7        'NA RADE, KDE BYL TISKNUT

```

POPIS.

```
450 FOR Y = Y1 TO Y2           'VYTvorit sloupce 10
    BODU VYSOKE.
460     LINE(X1,Y)-(X2,Y)
470     NEXT Y
480     R1 = R1 + 1
490     R2 = R2 + 2
500 NEXT K
510 *****
```

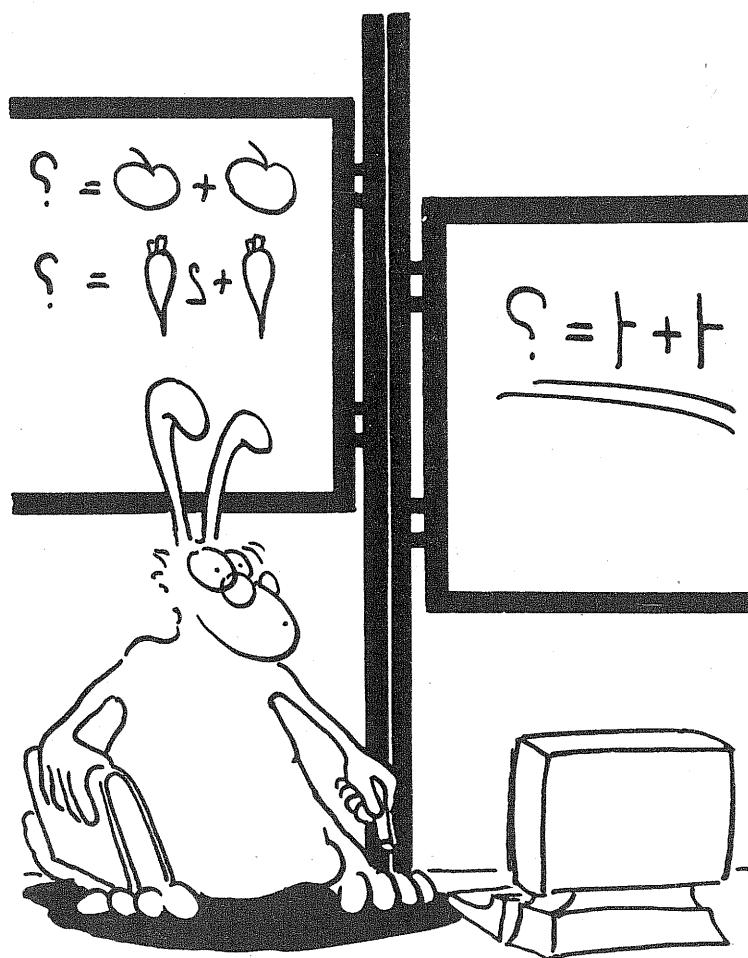
)

```
520 DATA TYDNY,0,25
530 DATA A,NAKUP VYBAVENI,0,2
540 DATA B,REKONSTRUKCE BUDOVY,0,16
550 DATA C,STANOVENI NAJMU,0,8
560 DATA D,INSTALACE VYBAVENI,2,8
570 DATA E,KOLAUDACNI KOMISE,16,3
580 DATA F,STANOVENI NAJEMNIKU,8,4
590 DATA G,SLAVNOSTNI OTEVRENI,19,3
600 END
```

Prog. 12-9 Časový diagram pro plánování úloh.



## 11. VZDĚLÁVACÍ GRAFIKA



Dostupnost laciných mikropočítačových grafických systémů poskytuje výkonný vzdělávací prostředek na všech stupních, od základních škol až po vysoké školy. Grafické programy můžeme vyvíjet pro demonstrace ve třídě nebo pro projekty samostudia v laboratoři. Takové programy pro počítačem řízenou výuku (CAI - Computer Assisted Instruction) mohou být všeobecně klasifikovány jako výcvikové (drill) a cvičící (practice), konzultační (tutorial) a dotazovací (inquiry), nebo jako simulační.

### **11.1. Výcvikové a cvičící programy**

Výcvikovými a cvičícími programy můžeme na stínítku obrazovky opakovaně prezentovat výukové problémy a požadovat odpovědi. Těmito problémy by mohly být dotazy na strukturu věty, cizí jazyky, historické osobnosti, umělecké formy, nebo geologická období. Odpovědi mohou být voleny z menu nebo napsáním. Výcvikové a cvičící programy můžeme navrhnout tak, aby na odpověď reagovaly zobrazením jednoduché zprávy, jako je "TO JE DOBRE" nebo "TO JE ŠPATNÉ. ODPOVĚZ ZNOVU.". Obvykle je však lépe, když program více pomáhá doplněním dalších informací při chybné odpovědi. Výcvikový program pro chemii o periodické tabulce prvků by se mohl ptát na atomové číslo náhodně vybraného prvku a reagovat na chybnou odpověď zprávou, který prvek, je-li takový, má uvedené atomové číslo. Pečlivěji vypracované programy by mohly reagovat na chybnou odpověď řadou "vodících" otázek nebo zobrazením obrázků a textu pomáhajících na cestě ke správné odpovědi.

Grafická zobrazení mohou pomáhat při formulaci mnoha problémů výcviku a cvičení. K výcviku hláskování je možno nakreslit obrázek předmětu jehož název má být hláskován (auto, loď, strom, dům) a otázky pro předmět ekonomie

mohou užívat grafy a diagramy. Tedy, aby bylo zobrazování programu zajímavější, můžeme obrázky doplnit k formulaci problému nebo do reakce na odpověď. Program 13-1, výcvik sčítání, užívá obrázky jak pro zábavu, tak pro vizuální formulaci předkládaného problému. Obrázek 13-1 ukazuje možné výstupy tohoto programu pro otázku  $4 + 5 = ?$ . Mají-li být reakce programu rozmanitější, můžeme náhodně vybírat různé tvary obličeje a textové fráze (DOBŘE, SKVĚLE, NE, VELMI ŠPATNÉ). Také bychom mohli volit různé tvary objektu pro každou formulaci problému.

```

10 'PROGRAM 13-1. ARITMETICKA CVICENI.
20     'POMOCI FUNKCE NAHODNEHO CISLA GENERUJE SCITACI
30     'PROBLEMY A PREDKLADA JE V TEXTOVE FORME
40     '(3 + 4 = ?). PREDKLADA PROBLEM JAKO SCITANI KRUZ-
50     'NIC, JSOU LI UVEDENY DVE CHYBNE ODPOVEDI. POKUD
60     'JE DALE UVEDENE CHYBNA ODPOVED (VICE NEZ DVE),
70     'POKRACUJEME DALSIM PROBLEMEM. UDRUZE SE POSET
80     '(R) SPRAVNYCH ODPOVEDI NA PRVY POKUS (KDYZ JE
90     'C = 1). PO 5 PROBLEMECH JE ZOBRAZENA SMEJICI SE
100    'TVAR A POCHVALNA ZPRAVA, JE-LI R > 3 (4 NEBO 5 Z
110    '5). PO 10 PROBLEMECH SE OBEVI DRUHY OBRAZEK A
120    'ZPRAVA, JE-LI R > 8.
130 ****
140 CLS
150 SCREEN 1
160 PRINT "AHOJI JAK SE JMENUJES";
170 INPUT N$
180 PRINT "DOBRA, "; N$; ", ZACNEME TEDY!"
190 PRINT "PRESVEDC SE KOLIK PROBLEMU SPRAVNE VYRESIS!"
200 FOR K = 1 TO 1000: NEXT K
210 ***** GENEROVANI 10 PROBLEMU *****
220 FOR P = 1 TO 10
230     C = 0   'C JE POSET ODPOVEDI NA PROBLEM
240     J = INT(RND(1) * 9 + .5)
250     K = INT(RND(1) * 9 + .5)
260     CLS
270     LOCATE 16,8
280     PRINT J; "+" ; K; "=" ;

```

```

290      INPUT A
300      C = C + 1
310      IF A = J + K THEN 490      'SPRAVNE ?
320      IF C <> 1 THEN 380
330      LOCATE 18,5          'ZODPOVEZENO 1x. ZNOVA
STEJNY POSTUP.
340      PRINT "TO NENI DOBRE, ";N$
350      PRINT "      ZKUS TO ZNOVU "
360      FOR F = 1 TO 1000: NEXT F
370      GOTO 260
380      IF C <> 2 THEN 410
390      GOSUB 610          'ZODPOVEZENO 2x. KRESLIT
KRUZNICE.
400      GOTO 270
410      IF C <> 3 THEN 460          'ZODPOVEZENO 3x. JESTE
JEDEN POKUS.
420      GOSUB 610
430      LOCATE 4,1
440      PRINT "ZKUS JESTE JEDNOU";
450      GOTO 270
460      LOCATE 20,1          'POKRACOVAT DALSIM
PROBLEMEM.
470      PRINT "ZKUSME DALSI";
480      FOR F = 1 TO 1000: NEXT F
490      IF C = 1 THEN R = R + 1      'SPRAVNE NA PRVY POKUS.
500      IF P = 5 AND R > 3 THEN GOSUB 790      'NAKRESLIT
SMEJICI SE TVAR
510      IF P = 10 AND R > 8 THEN GOSUB 960      'NAKRESLIT
BALONKY
520 NEXT P
530 '
540 CLS
550 LOCATE 10,19
560 PRINT "AHOJ,"
570 LOCATE 12,20-LEN(N$)/2
580 PRINT N$
590 GOTO 1260
600 '
610 ##### KRESLENI POMOCNYCH KRUZNIC #####
620 CLS
630 Y = 76

```

```

640 X = 8
650 FOR W = 1 TO J
660     CIRCLE(X,Y),5
670     X = X + 16
680 NEXT W
690 LOCATE 10,X/8+1
700 PRINT "+";
710 X = X + 20
720 FOR W = 1 TO K
730     CIRCLE(X,Y),5
740     X = X + 16
750 NEXT W
760 RETURN
770 '
780 ##### SMEJICI SE OBLICEJ #####
790 CLS
800 LOCATE 25,4
810 PRINT "ZATIM JE TO DOBRE, "; N$
820 CIRCLE(160,100),80
830 CIRCLE(130,80),5
840 CIRCLE(190,80),5
850 CIRCLE(160,100),3
860 FOR A = .79 TO 2.36 STEP 1/60  'KRESLENI UST OD 45 DO
    135 STUPNU
870     XP = 160 + 60 * COS(A)
880     YP = 80 + 60 * SIN(A)
890     PSET(XP,YP)
900 NEXT A
910 FOR F = 1 TO 1500: NEXT F
920 REM
930 RETURN
940 '
950 ##### BALONKY #####
960 CLS
970 LOCATE 4,17
980 PRINT N$
990 LOCATE 6,18
1000 PRINT "TO JE"
1010 LOCATE 8,17
1020 PRINT "OHROMNE!";
1030 CIRCLE(60,50),40

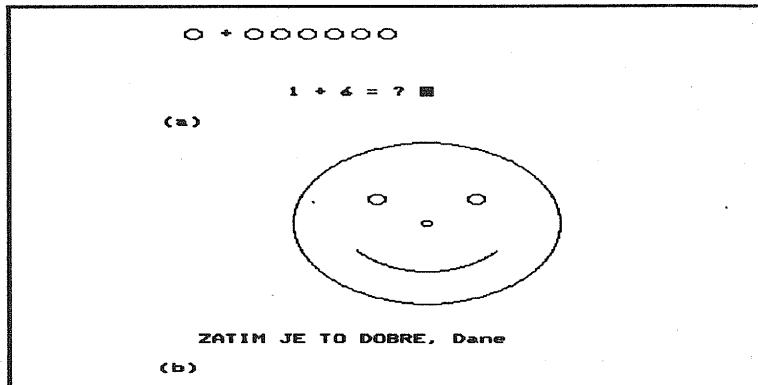
```

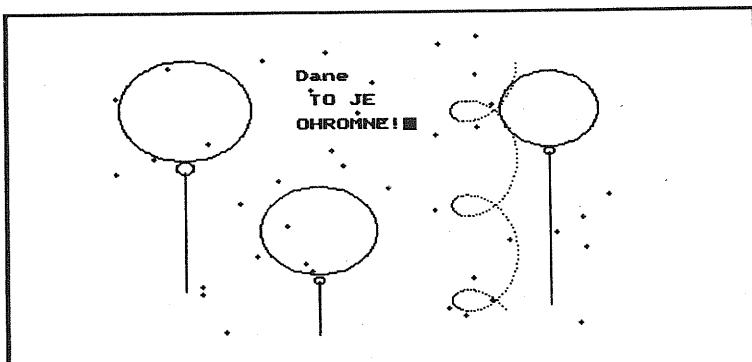
```

1040 CIRCLE(60,88),5
1050 LINE(60,91)-(60,170)      'PROVAZEK K BALONKU
1060 CIRCLE(280,50),30
1070 CIRCLE(280,78),3
1080 LINE(280,79)-(280,180)
1090 CIRCLE(140,130),35
1100 CIRCLE(140,163),3
1110 LINE(140,164)-(140,199)
1120 FOR Y = 2 TO 199 STEP 5      'VYTVORENI KONFET
1130     X = RND(1) * 319
1140     CIRCLE(X,Y),1
1150     FOR F = 1 TO 50: NEXT F
1160 NEXT Y
1170 X = 240: Y = 20: A = 0: R = 20: DA = 1/R/.5
1180 FOR Y = 20 TO 199           'VYTVORENI SPIRALY
1190     XP = X + R * COS(A)
1200     YP = Y + R * SIN(A)
1210     PSET(XP,YP)
1220     A = A + DA
1230 NEXT Y
1240 RETURN
1250 '#####
1260 END

```

Prog. 13-1 Cvičení aritmetiky, problémy sčítání jsou uváděny s návodem a obrázky.





(c)

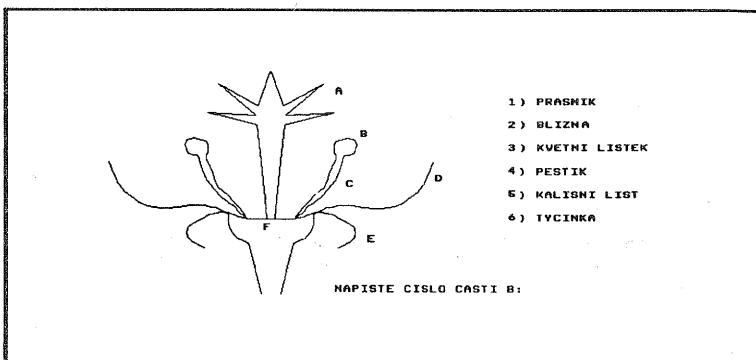
Obr. 13-1 Výstup programu Prog. 13-1 pro aritmetická cvičení, zobrazující (a) návodů při uvedení chybné odpovědi, (b) šťastný obličej pro řadu správných odpovědí a (c) balónky s provázky pro dobré konečné skóre.

## 11.2. Konzultační a dotazovací programy

V konzultačních a dotazovacích programech využíváme rozsáhlejší konverzační metody. Konzultační programy mohou poskytovat pokyny ke studiu a čtení určitých materiálů (knih, článků, filmů) pro samostudium a pak předkládat test z tohoto materiálu. Dotazovací programy mohou vést konverzaci odpovědí a dotazů. Například, dotazovací program může pomáhat při trénování zdravotních praktikantů odpovídáním na otázky, jako kdyby byl pacientem. Odpovědi programu by pak mohly být užity při tvorbě lékařské diagnózy. Zkoušky a otázky předkládané dotazovacími a konzultačními programy mohou používat grafiku stejně jako výcvikové a cvičící programy.

Pro programy samostudijního typu můžeme vytvářet diagramy částí rostlin a menu termínů, jako na Obr. 13-2.

Tento program vytváří několik takových zobrazení, vyhodnocuje odpovědi a dává známku za celkový výkon. Tento typ zobrazení můžeme použít jako součást zkušebního programu, nebo pro opakování a studium v konzultačním a dotazovacím programu.



Obr. 13-2 Obrázek zobrazovaný jako součást programu pro samostudium.

### 11.3. Simulační programy

Simulační programy používáme pro demonstraci chování různých typů systémů. Aktuální fyzikální nebo biologický model (nebo nějaký hypotetický systém) může být zobrazen v demonstraci lekce nebo jako součást programu pro individuální studium. Simulační programy jsou efektivní zvláště pro studium systémů s mnoha parametry, nebo systémů, které nemůžeme ve skutečnosti pozorovat. Modely geopolitických systémů, atomové a molekulární struktury, nebo relativistický pohyb objektů, jsou příklady takových systémů. Můžeme měnit parametry obsažené v těchto systémech a nastalé změny pozorovat na stínítku obrazovky.

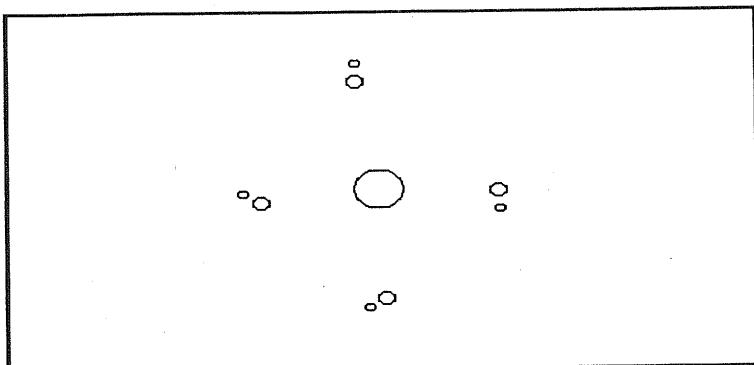
V mnoha grafických simulacích chceme demonstrovat složité pohyby, jako v Prog. 13-2. Zde vytváříme animovaný model sluneční soustavy. Obrázek 13-3 ukazuje několik poloh na dráze Měsíce a Země, jak se otáčejí kolem Slunce. Aktuální výstup Prog. 13-2 představuje pouze jednu polohu v čase, předcházející polohy jsou smazány. Pro zjednodušení jsou zde užity kruhové dráhy a objekty nejsou kresleny v měřítku.

Simulační programy mohou být navrženy jako vzdělávací hry. Střelecká hra, která vynáší dráhu koule podle výběru úhlu vrhu, může být efektivním vyučovacím programem. Tento program může pomoci porozumět geometrickému významu úhlových hodnot, stejně tak jako demonstrovat vztah mezi dráhou a úhlem vrhu

```
10  'PROGRAM 13-2. SLUNEČNÍ SOUSTAVA.
20  'ILUSTRUJE POHYB SLUNEČNÍ SOUSTAVY. MESIC OBIHA
30  'KOLEM ZEME 12 KRAT ZA KAZDOU JEDNU OTACKU ZEME
40  'KOLEM SLUNCE. DRAHA MESICE (RM) JE VOLENA JAKO
50  '1/6 DRAHY ZEME (RE) -- TAKZE PRIRUSTEK UHLU
60  '(1 DELENA POLOMEREM) JE SESTINASOBNY. UZITIM
70  'KROKU 1/RE A 1/RM BUDE MESIC CESTOVAT SE SESTINA-
80  'SOBNOU RYCHLOSTI RYCHLOSTI ZEME. VYNASIME NOVOU
90  'POLOHU MESICE DVAKRAT PRO KAZDOU ZEME, TAKZE
100 'MESIC SE BUDE POHYBOVAT 12 KRAT RYCHLEJI.
110 ****
120 CLS
130 XS = 128  'UMISTIT SLUNCE DO STREDU STINITKA 256 x 192
140 YS = 96
150 RE = 72    'RE JE POLOMER DRAHY ZEME
160 RM = 12    'RM JE POLOMER DRAHY MESICE
170 S = 15     'S JE POLOMER SLUNCE
180 E = 5      'E JE POLOMER ZEME
190 M = 3      'M JE POLOMER MESICE
200 SCREEN 1
210 CIRCLE(XS,YS),S   'KRESLIT SLUNCE
220 MO = 1/RM          'PRIRUSTEK UHLU PRO MESIC JE 1/RM
```

```
230 FOR EO = 1/RE TO 6.28318 STEP 1/RE 'PRO ZEMI JE 1/RE
240      REM
250      CIRCLE(XE,YE),E,0          'VYMAZANI ZEME
260      XE = XS + RE * COS(EO)
270      YE = YS + RE * SIN(EO)
280      REM
290      CIRCLE(XE,YE),E,1          'KRESLIT ZEMI
300      REM
310      CIRCLE(XM,YM),M,0          'VYMAZAT MESIC
320      XM = XE + RM * COS(MO)
330      YM = YE + RM * SIN(MO)
340      REM
350      CIRCLE(XM,YM),M,1          'KRESLIT MESIC
360      MO = MO + 1/RM
370      REM
380      CIRCLE(XM,YM),M,0          'VYMAZAT MESIC
390      XM = XE + RM * COS(MO)
400      YM = YE + RM * SIN(MO)
410      REM
420      CIRCLE(XM,YM),M,1          'KRESLIT MESIC
430      MO = MO + 1/RM
440 NEXT EO
450 GOTO 220
460 END
```

Prog. 13-2 Simulace: modelování sluneční soustavy s obíhajícím Měsícem a Zemí.



Obr. 13-3 Simulace pohybu Země a Měsíce kolem Slunce, ukazující několik poloh jako výstup Prog. 13-2.

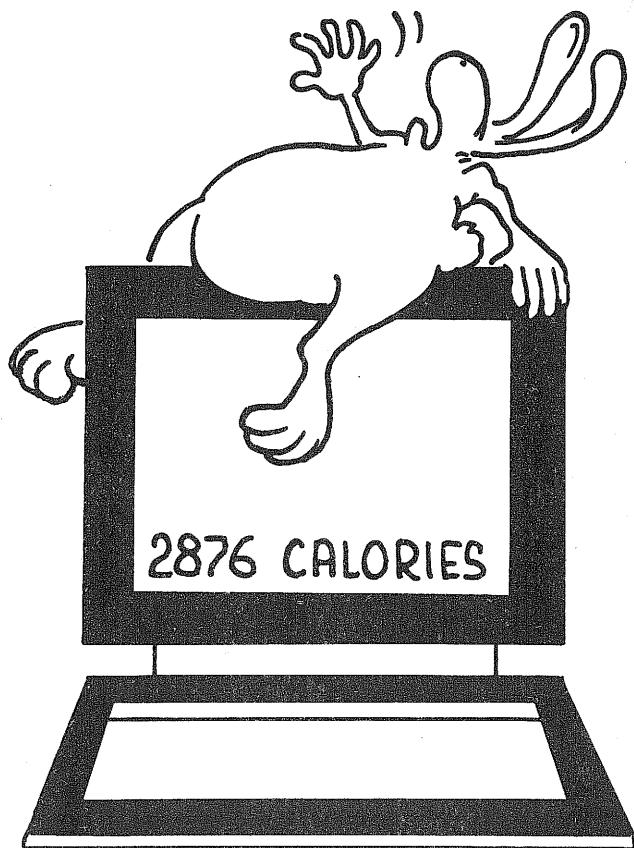
#### 11.4. Počítačem řízené vyučování

---

Grafické programy mohou být užity na pomoc při udržování záznamů a známkování. Programy CMI (Computer Managed Instruction) mohou být navrženy pro udržování záznamů o známkách, výpočet známeck a k vytváření statistik. Můžeme je používat pro vytváření grafů známeck pro jednu zkoušku, pro jeden kurs, pro určitý kurs za několik let, nebo pro všechny kurzy vyučované během libovolného časového intervalu.



## 12. OSOBNÍ GRAFIKA



Jako poslední v našem přehledu aplikací budeme diskutovat několik metod, jak využít grafiku, aby pracovala pro naše soukromé účely (osobní, personální grafika). Můžeme vytvářet osobní grafické programy pro účely rekreace, vzdělávání, nebo užitkové.

## 12.1. Grafika v domácím hospodaření

---

Existuje mnoho druhů finančních aplikací grafiky, které mohou být používány v domácnosti. Můžeme vynášet různé výdaje nebo sledované poplatky, analyzovat návraty investic, nebo vytvářet graf možných úspor, dosažených zlepšeným hospodařením s energií v domácnosti. Program 14-1 je příkladem měsíčního rozpočtu pro srovnávání výdajů. V tomto příkladu uvažujeme pouze čtyři kategorie výdajů (Obr. 14-1). Ve výsledném grafu jsou kresleny sloupce, ukazující procenta výdajů v každé kategorii za jeden měsíc. Podobně může být kresleno mnoho jiných výdajových grafů. Mohli bychom také sčítat týdenní a měsíční výdaje v souboru dat. Program pro rozpočet by pak mohl být použit pro srovnávání výdajů za několik měsíců nebo let.

```
10 'PROGRAM 14-1. CELKOVY ROZPOCET DOMACNOSTI.
20   'VYNASI SLOUPCOVY DIAGRAM MESICNIH VYDAJU.
30   'VYNASENE SLOUPCE JSOU PROCENTEM CELKOVYCH VYDAJU
40   'ZA MESIC. TAKE TISKNE SKUTECNE VYDAJE V KAZDE
50   'KATEGORII -- STRAVA, OBLEKANI, BYDLENI, REKREACE.
60   'VELIKOST STINITKA JE 320 x 200 BODU.
70   ***** VSTUP DAT *****
80 CLS
90 D = INT(290 / 4)      'CTYRI KATEGORIE VYDAJU
100 RS = (172 - 12)       'RS JE ROZSAH BODU PRO SLOUPCE
110 T = 0                  'T JE SOUCET VSECH VYDAJU
120 PRINT "UVED NAZEV MESICE"
130 INPUT M$
140 PRINT
150 PRINT "F - STRAVA      H - BYDLENI      C - OBLEKANI      R -
```

```

REKREACE"
160 PRINT
170 PRINT "UVED VYDANI A KOD KATEGORIE. UVED 0,Q PRO
UKONCENI"
180 INPUT E, C$
190 IF E = 0 AND C$ = "Q" THEN 300
200 IF C$ = "F" OR C$ = "H" OR C$ = "C" OR C$ = "R" THEN
230
210 PRINT "CHYBNY KOD VYDANI"
220 GOTO 180
230 T = T + E
240      'SCITANI VYDAJU PRO KAZDOU KATEGORII
250 IF C$ = "F" THEN FT = FT + E
260 IF C$ = "H" THEN HT = HT + E
270 IF C$ = "C" THEN CT = CT + E
280 IF C$ = "R" THEN RT = RT + E
290 GOTO 180
300      ***** KRESLENI SOURADNE SITE A POPISU *****
310 CLS
320 SCREEN 1
330 LINE(25,12)-(25,172)
340 LINE(25,172)-(315,172)
350 LOCATE 1,1      'TISK NAZVU MESICE
360 P = 20 - LEN(M$) / 2      'CENTROVANI NAZVU MESICE
370 PRINT TAB(P);M$
380 RO = 22
390 Y = 172
400 FOR K = 0 TO 4      'POPIS VERTIKALNI OSY SOUSEDNIMI
410     LOCATE RO,1      'PETINAMI ROZSAHU
420     L = LO + 100 * K / 5
430     PRINT USING "###";L
440     RO = RO - 4
450     Y = Y - 32
460 NEXT K
470 LOCATE 2,1
480 PRINT "PROCENT";
490      'POPIS ODDILU
500 LOCATE 23,1
510 PRINT "      STRAVA BYDLENI OBLEKANI REKREACE";
520      ***** VYTVORENI SLOUPCU *****
530 YO = 171

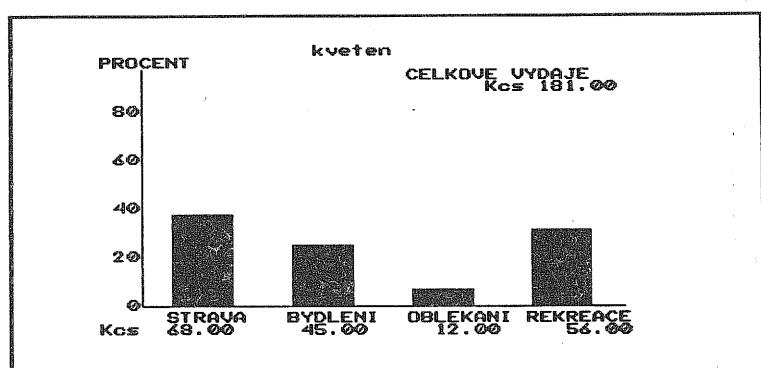
```

```

540 X1 = 25 + D/4      'ZAHAJIT PRVY SLOUPEC O 1/4 DALE V
    PRVEM ODDILU
550 Y = INT((1 - FT/T) * RS + 12.5)   'VYJADRENI NAKLADU
560 GOSUB 700                 'KAZDE KATEGORIE V %
570 Y = INT((1 - HT/T) * RS + 12.5)   'CELK. SOUCTU, ODECIST
580 GOSUB 700                 'OD 1 (TAK PUJDOU
590 Y = INT((1 - CT/T) * RS + 12.5)   'SLOUPCE NAHORU) A NA-
600 GOSUB 700                 'SOBIT ROZSAHEM BODU
610 Y = INT((1 - RT/T) * RS + 12.5)   'UZITYM PRO SLOUPCE.
620 GOSUB 700
630 LOCATE 24,1
640 PRINT USING "Kcs###.##     ###.##     ###.##"
    ##.##";FT;HT;CT;RT;
650 LOCATE 3,24
660 PRINT "CELKOVE VYDAJE";
670 LOCATE 4,30
680 PRINT USING "Kcs###.##";T
690 GOTO 770
700 '##### VYTVORENI SLOUPCE #####
710 FOR X = X1 TO X1 + D / 2
720     LINE(X,Y)-(X,Y0)
730 NEXT X
740 X1 = X1 + D
750 RETURN
760 '#####
770 END

```

Prog. 14-1 Sloupcový diagram domácího rozpočtu.



Obr. 14-1 Zobrazení domácích výdajů programem Prog. 14-1.

Grafy a diagramy můžeme používat pro různé nefinanční účely. Denní diagram výživy je uveden na Obr. 14-2. Tento diagram je výstupem Prog. 14-2, který počítá denní kalorický příjem a vynáší procenta kalorií z bílkovin, uhlohydrátů a tuků v naší stravě. Tuto denní informaci bychom mohli ukládat do datového souboru a upravit Prog. 14-2 pro vynášení dlouhodobých srovnání. Pro takový program by mohl být také sestaven obsáhlější soubor typů a složení jídel.

```

10  'PROGRAM 14-2. NUTRICNI ANALYZA.
20  'CTE JMENA JIDEL A ODPOVIDAJICI HODNOTY Z PRIKAZU
30  'DATA DO POLI. CTE POCET GRAMU TUKU, UHLOHYDRATU,
40  'BILKOVIN, DRASLIKU A SODIKU PRO PRUMERNOU VELI-
50  'KOST POKRNU. UVADIME JMENA JIDEL, KTERA JSME ZA
60  'DEN SNEDLI. GRAMY TUKU, BILKOVIN, ATD. JSOU SCI-
70  'TANY PRO VSECKA RUZNA JIDLA. GRAMY BILKOVINY,
80  'TUKU A UHLOHYDRATU JSOU PREVEDENY NA CELKOVY PO-
90  'CET SPOTREBOVANYCH KALORII. KOLACOVY DIAGRAM UKA-
100  'ZUJE PROCENTA CELKOVYCH KALORII Z UHLOHYDRATU,
110  'TUKU A BILKOVIN. TAKE KRESLI DVA SLOUPCOVE DIA-
120  'GRAMY -- JEDEN UKAZUJE POCET SPOTREBOVANYCH GRAMU
130  'TUKU, UHLOHYDRATU A BILKOVINY; DRUHY UKAZUJE

```

```

135      'POCET GRAMU DRASLIKU A SODIKU. PREDPOKLADANA
140      'VELIKOST STINITKA OBRAZOVKY 320 x 200 BODU.
145      'SLOUPCE JSOU ZOBRAZENY DO 12 - 172.
150      ****
160  CLS
170  DIM F$(10),FA(10),CA(10),PR(10),PO(10),SO(10)
180  YA = 5/6
190  FOR K = 1 TO 10
200      READ F$(K),FA(K),CA(K),PR(K),PO(K),SO(K)
210  NEXT K
220      **** VSTUP POPISU A DAT SLOUPCU ****
230  PRINT "UVED JMENA VSECH JIDEL. UVED HOTOVO PRO KONEC"
240  INPUT F1$
250  IF F1$ = "HOTOVO" THEN 370
260  FOR K = 1 TO 10          'NALEZENI JMENA JIDLA V POLI
270      IF F1$ = F$(K) THEN 310
280  NEXT K
290  PRINT "NEBYLO NALEZENO. UVED ZNOVU NEBO JMENO DALSIHO
         JIDLA"
300  GOTO 240
310  FAT = FAT + FA(K)          'SCITANI SLOZEK JIDLA
320  CAT = CAT + CA(K)
330  PRT = PRT + PR(K)
340  POT = POT + PO(K)
350  SOT = SOT + SO(K)
360  GOTO 240
370      **** KRESLENI SOURADNE SITE A POPISU ****
380  CLS
390  SCREEN 1
400  H1 = PRT          'NALEZT NEJVETSII HODNOTU PRO
         PRVY SLOUPCOVY DIAGRAM
410  IF FAT > H1 THEN H1 = FAT
420  IF CAT > H1 THEN H1 = CAT
430  R1 = (172 - 12) / H1      'R1 JE MERITKO PRO PRVY
         DIAGRAM
440  H2 = POT          'URCIT NEJVETSII HODNOTU PRO
         DRUHY SLOUPCOVY DIAGRAM
450  IF SOT > H2 THEN H2 = SOT
460  R2 = (172 - 12) / H2      'R2 JE MERITKO PRO DRUHY
         DIAGRAM
470  LINE(256,12)-(256,172)

```

```

480 LINE(256,172)-(319,172)
490 LINE(144,12)-(144,172)
500 LINE(144,172)-(214,172)
510 RO = 22
520 Y = 172
530 FOR K = 0 TO 5           'POPIS VERTIKALNI OSY SOUSED-
540   LOCATE RO,17          'NIMI PETINAMI ROZSAHU
550   L = H1 * K / 5
560   PRINT USING "###";L;
570   LOCATE RO,31
580   L = H2 * K / 5
590   PRINT USING ".##";L;
600   RO = RO - 4
610   Y = Y - 32
620 NEXT K
630 LOCATE 1,18
640 PRINT "GRAMU      GRAMU";
650   'POPIS ODDILU
660 LOCATE 23,21
670 PRINT "F C P      P S";
680 LOCATE 24,21
690 PRINT "A A P      O O";
700 LOCATE 25,21
710 PRINT "T R O      T D";
720 '
730   ***** VYTVORENI SLOUPCU *****
740 Y0 = 171
750 X1 = 160
760 Y = INT((H1 - FAT) * R1 + 12.5)
770 C=2          'MODRA
780 GOSUB 910      'VYTVORENI SLOUPCE
790 Y = INT((H1 - CAT) * R1 + 12.5)
800 C=5          'ZLUTA
810 GOSUB 910      'VYTVORENI SLOUPCE
820 Y = INT((H1 - PRT) * R1 + 12.5)
830 C=4          'ZELENA
840 GOSUB 910      'VYTVORENI SLOUPCE
850 X1 = 280        'PRESUN NA DRUHY SLOUPCOVY
     DIAGRAM
860 Y = INT((H2 - SOT) * R2 + 12.5)
870 GOSUB 910      'VYTVORENI SLOUPCE

```

```

880 Y = INT((H2 - POT) * R2 + 12.5)
890 GOSUB 910          'VYTVORENI SLOUPCE
900 GOTO 980
910      'VYTVORENI SLOUPCE
920 FOR X = X1 TO X1 + 8
930   LINE(X,Y)-(X,Y0),C
940 NEXT X
950 X1 = X1 + 16          'PRESUN NA DALSI SLOUPEC
960 RETURN
970 '
980 ##### VYTVORENI KOLACOVEHO DIAGRAMU #####
990 XC = 45
1000 YC = 78
1010 R = 40
1020 FAC = FAT * 9          'PREVOD GRAMU NA KALORIE
1030 CAC = CAT * 4
1040 PRC = PRT * 4
1050 CAL = FAC + CAC + PRC  'URCENI KALORII CELKEM
1060 LOCATE 3,1
1070 PRINT "KALORII - ";
1080 LOCATE 4,3
1090 PRINT USING "####";CAL;
1100 LOCATE 17,1
1110 PRINT "% Z CELKU";
1120 LOCATE 18,1
1130 PRINT " Z -";
1140 S = 0
1150 B = 0
1160 S = S + FAC
1170 C=2          'MODRA
1180 GOSUB 1320          'VYSEC PRO KALORIE TUKU
1190 S = S + CAC
1200 C=5          'ZLUTA
1210 GOSUB 1320          'VYSEC PRO KALORIE UHLOHYDRATU
1220 S = S + PRC
1230 C=4          'ZELENA
1240 GOSUB 1320          'VYSEC PRO KALORIE BILKOVIN
1250 LOCATE 20,1
1260 PRINT "TUK";TAB(10);USING "##";FAC/CAL*100;
1270 LOCATE 21,1
1280 PRINT "UHLOH";TAB(10);USING "##";CAC/CAL*100;

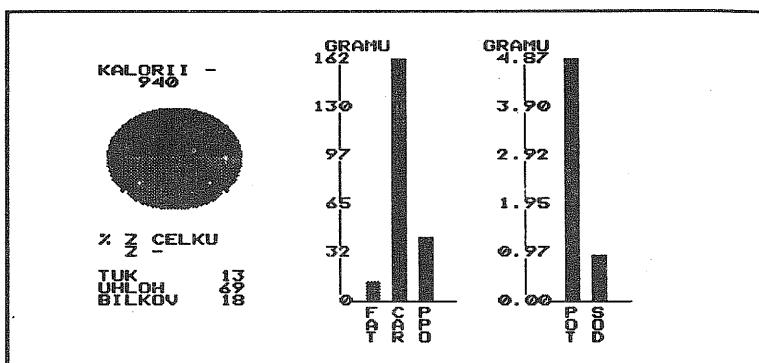
```

```

1290 LOCATE 22,1
1300 PRINT "BILKOV";TAB(10);USING "##";PRC/CAL*100;
1310 GOTO 1540
1320 '#####
1330 DA = 1/R/50          'MALY KROK PRO SOUVISLE
    VYPLNENI KRUZNICE BARVOU
1340 A = 6.28318 * S / CAL  'A JE UHEL KONCE TOHOTO REZU
1350 FOR A1 = B TO A STEP DA  'B JE UHEL KONCE POSLEDNIHO
    REZU
1360      XP = XC + R * COS(A1)
1370      YP = YC + R * SIN(A1) * YA
1380      LINE(XP,YP)-(XC,YC),C
1390 NEXT A1
1400 B = A
1410 RETURN
1420 '#####
1430 DATA MLEKO,12,9,9,.122,.351
1440 DATA SLANINA,1,8,5,.163,.038
1450 DATA TRESKA,5,6,20,.177,.348
1460 DATA TUNAK,0,18,21,.688,.259
1470 DATA VEJCE,0,6,7,.066,.070
1480 DATA SPENAT,3,0,2,.040,.259
1490 DATA KUKURICE,1.4,26.32,4.48,0,.231
1500 DATA DATLE,.89,129,3.9,.002,1.15
1510 DATA JATRA,.6,.3,1.5,.022,3.04
1520 DATA PAPRIKA,5.2,10.4,6.4,.001,.45
1530 '#####
1540 END

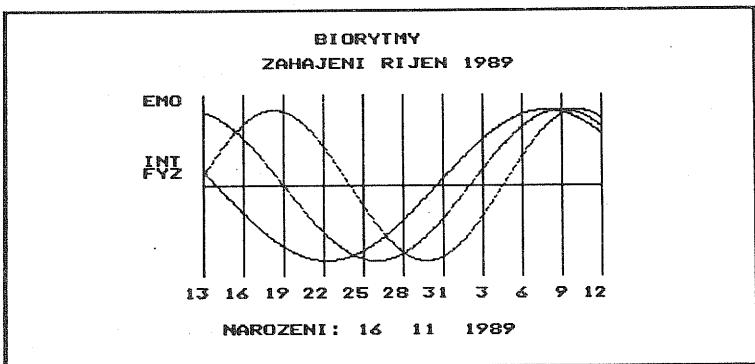
```

Prog. 14-2 Graf výživy (nutriční graf), vynášející kalorie a složky stravy do dvou sloupcových diagramů a do diagramu kruhových výsečí.



Obr. 14-2 Nutriční diagram vytvořený programem Prog. 14-2

Program 14-3 vytváří graf biorytmů, jak je uvedeno na Obr. 14-3. Tento graf vynáší teoretické vzrůsty a poklesy našich fyzických, emocionálních a intelektuálních energetických hladin. Algoritmus, užity pro tento graf, předpokládá 23-denní cyklus pro fyzickou křivku, 28-denní cyklus pro emocionální křivku a 33-denní cyklus pro intelektuální křivku.



Obr. 14-3 Graf biorytmů zobrazený Prog. 14-3 pro datum narození 4.8.1963.

```

10  'PROGRAM 14-3. BIORYTMY
20  'PO UVEDENI DATA NAROZENI A DATA ZACATKU VYTvorI
30  'DIAGRAM BIORYTMU (S KRIVKAMI PRO EMOCIONALNI,
40  'FYZICKY A INTELEKTUALNI CYKLUS) PRO PRISTICH 30
50  'DNU. KRIVKY JSOU KRESLENY S POPISY, KTERE JSOU
60  'UMISTENY VLEVO, POKUD NEZACINAJI LIB. KRIVKY
70  'PRILIS BLIZKO SEBE -- PAK JE 1 Z POPISU PREMISTEN
80  'NA PRAVOU STRANU. DIAGRAMU. PREPOKLADANA VELIKOST
90  'STINITKA JE 320 x 200 BODU.
100 *****

110CLS
120DIM M$(12), DM(12), AD(13)
130FOR K = 1 TO 12      'NACTENI JMENA MESICE, # DNE V MESICI
140    READ M$(K), DM(K), AD(K)  'A SOUCET DNI
150NEXT K
160PR = 8                'PR JE POSET BODU NA ZNAK VERTIKALNE
170PI = 3.14159
180H = 50                 'H JE VYSKA KRIVEK
190RR = (284 - 44) / 30   'RR JE MERITKO: 30 DNU NA 44-284
200'
210***** VSTUP DATUMU *****
220PRINT "UVED DATUM NAROZENI (MESIC, DEN, ROK)"
230INPUT MB, DB, YB
240PRINT "UVED PRVE DATUM DIAGRAMU (MESIC, DEN, ROK)"
```

```

250 INPUT MC,DC,YC
260      'URCENI CASU MEZI DATEM DIAGRAMU A DATEM NAROZENI
270 T = (YC-YB) * 365.25 + (AD(MC) - AD(MB)) + (DC - DB)
280      'URCENI PRIRUSTKU PRO KAZDOU KRIVKU
290 PD = 2 * PI * (T / 23 - INT(T / 23))
300 ED = 2 * PI * (T / 28 - INT(T / 28))
310 ID = 2 * PI * (T / 33 - INT(T / 33))
320      'URCENI FREKVENCE PRO KAZDOU KRIVKU
330 PW = 2 * PI / 23
340 EW = 2 * PI / 28
350 IW = 2 * PI / 33
360      '
370      ***** KRESLENI SOURADNE SITE *****
380 CLS
390 SCREEN 1
400 LOCATE 1,15
410 PRINT "BIORYTMY"
420 LOCATE 3,20-(13+LEN(M$(MC)))/2
430 PRINT "ZAHAJENI "; M$(MC); YC;
440 D = DC
450 X = 44
460 FOR CO = 5 TO 35 STEP 3
470      LOCATE 22,CO
480      PRINT USING "##";D;
490      D = D + 3
500      IF D <= DM(MC) THEN 520      'JESTE STENY MESIC ?
510      D = D - DM(MC)           'NASTAVIT D NA ZACATEK
      DALSIHO MESICE
520      LINE(X,40)-(X,160)
530      X = X + 24
540 NEXT CO
550 LINE(44,100)-(284,100)      'KRESLENI USECKY NA 0
560 LOCATE 25,8
570 PRINT "NAROZENI:";DB;MB;YB;
580      ***** KRESLENI KRIVEK *****
590 FOR X = 0 TO 30 STEP .1
600      PY = INT(H * SIN(PW * X + PD) + .5)
610      XG = X * RR + 44          'TRANSFORMACE X DO
      SOURADNICE GRAFU
620      PSET(XG,PY+100)
630      EY = INT(H * SIN(EW * X + ED) + .5)

```

```

640      PSET(XG,EY+100)
650      IY = INT(H * SIN(IW * X + ID) + .5)
660      PSET(XG,IY+100)
670      IF X = 0 THEN GOSUB 780      'PRECHOD NA VYTVORENI
          POPISU
680      NEXT X
690      IF EL$ <> "ANO" THEN 730  'JE NUTNO KRESLIT NEJAKY
          POPIS VPROAVO ?
700      EP = INT((EY + 100) / PR + .5)
710      LOCATE EP,38
720      PRINT "EMO";
730      IF IL$ <> "ANO" THEN 770
740      IP = INT((IY + 100) / PR + .5)
750      LOCATE IP,38
760      PRINT "INT";
770      GOTO 970
780      ##### POPISOVANI VLEVO #####
790      PP = INT((PY + 100) / PR + .5)  'TISK. POZICE PRO
          FYZICKY
800      LOCATE PP,2
810      PRINT "FYZ";
820      EP = INT((EY + 100) / PR + .5)  'TISK. POZICE PRO
          EMOCIONALNI
830      IF EP = PP THEN EL$ = "ANO"      'BUDE POTREBA POPIS
          UMISTIT VPROAVO ?
840      IF EL$ = "YES" THEN 870
850      LOCATE EP,2
860      PRINT "EMO";
870      IP = INT((IY + 100) / PR + .5)  'TISK. POZICE PRO
          INTELEKTUALNI
880      IF IP = PP OR IP = EP THEN IL$ = "ANO"      'BUDE
          POTREBA POPIS UMISTIT VPROAVO ?
890      IF IL$ = "YES" THEN 920
900      LOCATE IP,2
910      PRINT "INT";
920      RETURN
930      ****
940      DATA LEDEN,31,0,UNOR,28,31,BREZEN,31,59,DUBEN,30,90
950      DATA KVETEN,31,120,CERVEN,30,151,CERVENEC,31,181
960      DATA SRPEN,31,212,ZARI,30,243,RIJEN,31,273
970      DATA LISTOPAD,30,304,PROSINEC,31,334

```

980 END

Prog. 14-3 Graf biorytmů.

Počítačem vytvářené obrázky mohou být určeny pro mnoho typů domácího využití. Obrázky můžeme vkládat do grafů, do vzdělávacích programů, jako součást her, nebo pro dekoraci. Tištěný obrázek může být využit jako dekorace na stěnu nebo jako osobní navržené gratulační pohlednice. Také můžeme vytvářet malé obrázky nebo návrhy pro osobní dopisní papíry.

## 12.2. Hraní her

---

Hry na počítači mohou být zábavné i vzdělávací. Můžeme sestavit programy pro hry, které učí matematiku, aritmetiku, písmena, slova, nebo hláskování, dynamické hry mohou napomáhat rozvoji koordinace a pod..

Program 14-4 simuluje hru s míčkem a pálkou, která vyžaduje určitou koordinaci, má-li se míček udržet ve hře. Tento program používá obdélník a skákající míček a doplňuje páalku místo levé stěny. Míček začíná v náhodné poloze uvnitř obdélníka. Kdykoliv se dostane na levou stranu, musíme ho odrazit zpět do obdélníka. Pokud jej mineme, vylétne ven z obdélníka a zastaví se. Počet, kolikrát po sobě jsme odrazili míček pálkou, je naše skóre. Po každých pěti odrazech od páalky se míček poněkud zrychlí. Obrázek 14-4 ukazuje zobrazení na konci hry pro jedno možné skóre. Tuto hru můžeme hrát sami a sledovat, jak vysokého skóre dosáhneme v jedné hře, nebo můžeme hrát ve skupinách se sčítáním bodů, dokud jedna ze skupin nedosáhne 100 a pod.

```
10  'PROGRAM 14-4. HRA SE SKAKAJICIM MICKEM.  
20      'KRESLI OBDELNIK SE TREMI STRANAMI A PALKU NA LEVE  
30      'STRANE. MICEK SE ODRAZI UVNITR OBDELNIKA (DX =  
40      'DY = 5 NA ZACATKU HRY) A OD PALKY. PALKA MUSI BYT
```

```

50      'UMISTENA POMOCI KLAVES ("U" = NAHORU, "D" = DOLU).
60      'RYCHLOST MICKU VZRUSTA (ZVYSOVANIM DX A DY) KDYZ
70      'SE MICEK ODRAZI 5 x OD PALKY STEJNOU RYCHLOSTI.
80      'SKORE JE ZVYSENO O 1 PRO KAZDY USPESNY ODRAZ OD
90      'PALKY. HRA POKRACUJE DOKUD MICEK NEMINE PALKU.
100     '*****
110     CLS
120     XL = 50      'XL,XR,YT,YB JSOU HRANICE OBDELNIKA
130     XR = 300
140     YT = 20
150     YB = 160
160     YP = 160      'YP JE HORNI BOD PALKY
170     R = 3      'R JE POLOMER MICKU
180     DX = 5      'MICEK SE ZPOCATKU POSOUVA O 5 JEDNOTEK
190     DY = 5      'V KAZDEM KROKU
200     SCREEN 1
210     '***** KRESLENI OBDELNIKA *****
220     LINE(XL,YT)-(XR,YT)
230     LINE(XR,YT)-(XR,YB)
240     LINE(XR,YB)-(XL,YB)
250     LINE(XL,YP)-(XL,YP+40)      'KRESLENI PALKY
260     '***** ODRAZEJICI SE MICEK *****
270     XN = XL + INT((XR-XL) / 2)      'MICEK ZACINA UPROSTRED
280     YN = YT + INT((YB-YT) / 2)      'OBDELNIKA
290     S = 0      'S = OKAMZITE SKORE
300     LOCATE 1,36
310     PRINT USING "#";S
320     REM
330     CIRCLE(X,Y),R,0      'VYMAZAT MICEK V PUvodni
340     REM      'POLOZE.
350     CIRCLE(XN,YN),R,1      'KRESLIT V NOVE POLOZE
360     'PRESUN PALKY
370     A$ = INKEY$
380     IF A$ = "" THEN 450
390     REM COLOR 0,0
400     LINE(XL,YP)-(XL,YP+40),0      'VYMAZAT PALKU V PUV.POL.
410     IF A$ = "D" THEN YP = YP + 15
420     IF A$ = "U" THEN YP = YP - 15
430     REM
440     LINE(XL,YP)-(XL,YP+40),1      'KRESLIT PALKU V NOVE POL.
450     X = XN      'ULOZENI AKTUALNI POLOHY DO X A Y

```

```

460 Y = YN
470 BX = 0           'BX A BY JSOU INDIKATORY URCUJICI
480 BY = 0           'OD KTERE STENY SE BUDE ODRAZET
490 M = DY / DX      'M JE SMERNICE DRAHY MICKU
500 '
510 *****+
520 'UDERIME DO VERTIKALNI STENY?
530 IF DX > 0 THEN 600 'PUJDENE VPRAVO
540 'JINAK PUJDENE VLEVO (DX JE ZAPORNE)
550 IF X + DX - R > XL THEN 650 'JSME JESTE UVNITR OBDEL.
560 BX = 1           'JDE VEN Z OBDEL. U STENY X
570 XN = XL + R + 1   'NOVE X JE PRAVE UVNITR LEVE STENY
580 GOTO 650
590 '++++++*
600 'POKRACOVANI VPRAVO
610 IF X + DX + R < XR THEN 650 'JSME JESTE V OBDELNIKU
620 BX = 1           'JDE VEN Z OBDEL. U STENY X
630 XN = XR - R - 1   'NOVE X JE PRAVE UVNITR PRAVE STENY
640 *****+
650 'UDERIME DO HORIZONTALNI STENY?
660 IF DY > 0 THEN 730 'PUJDENE DOLU
670 'JINAK PUJDENE NAHORU(DY JE ZAPORNE)
680 IF Y + DY - R > YT THEN 790 'JSME JESTE UVNITR OBDEL.
690 BY = 1           'JDE VEN Z OBDEL. U STENY Y
700 YN = YT + R + 1   'NOVE Y JE PRAVE UVNITR HORNÍ STENY
710 GOTO 790
720 '++++++*
730 'POKRACOVANI DOLU
740 IF Y + DY + R < YB THEN 790 'JSME JESTE V OBDELNIKU
750 BY = 1           'JDE VEN Z OBDEL. U STENY Y
760 YN = YB - R - 1   'NOVE Y JE PRAVE UVNITR DOLNI STENY
770 '
780 *****+
790 'ODRAZI SE OD ZADNE STENY, STENY X, STENY Y, NEBO
    OD OBOU STEN?
800 IF BX = 0 AND BY = 0 THEN 850 'NEODRAZI SE
810 IF BX = 0 AND BY = 1 THEN 890 'ODRAZI SE OD Y
820 IF BX = 1 AND BY = 0 THEN 930 'ODRAZI SE OD X
830 IF BX = 1 AND BY = 1 THEN 1060 'ODRAZI SE OD OBOU
840 '          (V ROHU)
850 ##### NEODRAZI SE #####

```

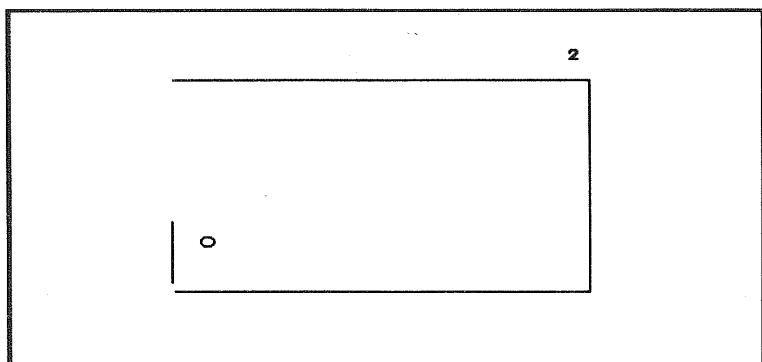
```

860 XN = X + DX
870 YN = Y + DY
880 GOTO 330
890 '***** ODRAZI SE OD STENY Y *****
900 XN = (YN - Y) / M + X
910 DY = -DY
920 GOTO 330
930 '***** ODRAZI SE OD STENY X *****
940 YN = (XN - X) * M + Y
950 IF DX > 0 THEN 1040      'ODRAZI SE VPRAVO
960      'JDE PROTI PALCE. JE PALKA VE SPRAVNE POLOZE ?
970 IF YN < YP OR YN > YP+40 THEN 1140 'MICEK MINUL PALKU
980 S = S + 1                  'ZVYSENI SKORE
990 LOCATE 1,36
1000 PRINT USING "##";S;
1010 IF S/5 <> INT(S/5) THEN 1040    'JE-LI ROVNO, JE CAS
      ZRYCHLIT MICEK
1020 IF DX < 0 THEN DX = DX - 5
1030 IF DX > 0 THEN DX = DX + 5
1040 DX = -DX
1050 GOTO 330
1060 '***** ODRAZI SE V ROHU *****
1070      'DO KTERE STENY BY NARAZIL DRIVE ?
1080 IF ABS(XN - X) < ABS(YN - Y) THEN 930  'ODRAZ OD X
1090 IF ABS(YN - Y) < ABS(XN - X) THEN 890  'ODRAZ OD Y
1100      'MIC JE STEJNE DALEKO OD STENY X I Y V ROHU
1110 DX = -DX
1120 DY = -DY
1130 GOTO 330
1140 '***** MICEK MINUL STENY I PALKU *****
1150 XN = XL - R - 8  'SPADNUTI MICKU DOLU NA ZEM VEDLE
1160 YN = (XN - X) * M + Y  'LEXE STRANY OBDELNIKA
1170 REM
1180 CIRCLE(X,Y),R,0
1190 FOR YN = YN TO 190 STEP 5
1200      REM
1210      CIRCLE(X,Y),R,0
1220      REM
1230      CIRCLE(XN,YN),R,1
1240      FOR J = 1 TO 1000: NEXT J
1250      Y = YN

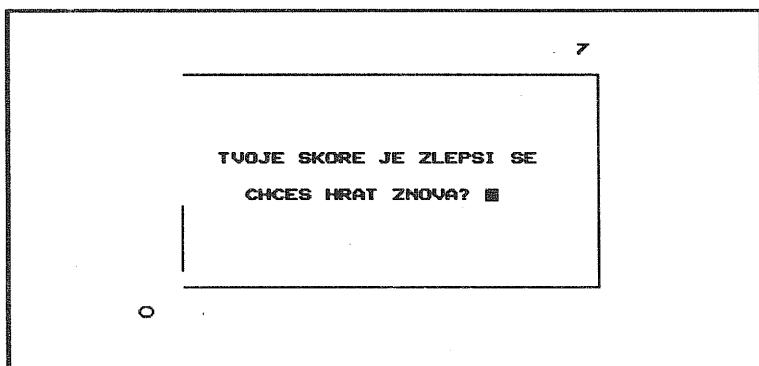
```

```
1260      X = XN
1270 NEXT YN
1280 ***** TISK SKORE *****
1290 IF S < 24 THEN 1320
1300 Z$ = "NEUVERITELNE"
1310 GOTO 1450
1320 IF S < 19 THEN 1350
1330 Z$ = "VYNIKAJICI!"
1340 GOTO 1450
1350 IF S < 14 THEN 1380
1360 Z$ = "VELMI DOBRE"
1370 GOTO 1450
1380 IF S < 9 THEN 1410
1390 Z$ = "UJDE TO"
1400 GOTO 1450
1410 IF S < 4 THEN 1440
1420 Z$ = "ZLEPSI SE"
1430 GOTO 1450
1440 Z$ = "STRASNE"
1450 LOCATE 10,10
1460 PRINT "TVOJE SKORE JE ";Z$
1470 LOCATE 13,12
1480 ***** HRAT ZNOVA NEBO SKONCIT ? *****
1490 PRINT "CHCES HRAT ZNOVA";
1500 INPUT C$
1510 IF C$ = "A" THEN 180
1520 END
```

Prog. 14-4 Hra se skákJícím míčkem a pálkou.



(a)



(b)

Obr. 14-4 Dvě zobrazení hry míčku a páinky z Prog. 14-4.

Naším posledním příkladem je lukostřelecká hra v Prog. 14-5. Obdélník je umístěn do náhodné polohy v pravé polovině stínítka a my se pokoušíme jej trefit šípem střeleným z levého rohu. Volíme úhel a počáteční rychlosť a šíp letí po

parabolické dráze, jak jsme diskutovali v předchozí kapitole. Máme tři rány z toulce s šípy (Obr. 14-5) na každý obdélník. Program generuje pro každou hru pět obdélníků, zobrazuje číslo obdélníka a celkové skóre. Počítáme trefu na první ránu jako 10 bodů, na druhou ránu jako 5 bodů a na třetí za 2 body.

U mnoha mikropočítáčů můžeme doplnit do programů zvukový výstup. Zvuky využíváme v našich hrách tak, aby odpovídaly odrazu míčku nebo dopadu šípu. Na konci hry můžeme dokonce zahrát melodii. Generování zvuků může zpomalit zpracování, pokud nemá systém schopnost produkovat zvuk současně s jiným zpracováním.

```
10  'PROGRAM 14-5. HRA SIP S TERCEM.
20  'OPAKOVANE KRESLI A MAZE SIP, JEHOZ KONEC
30  'JE BODEM NA PARABOLE. ZBYTEK SIPU JE NALEZEN
40  'PODLE TOHOTO KONCOVEHO BODU, SMERNICE TANGENTY
50  'PARABOLY V TOMTO BODU A DELKY SIPU.
60  '*****                                 *****
70  '
80 XM = 319
90 YM = 199
100 XO = 15      'XO,YO JE POCATECNI POLOHA SIPU
110 YO = 180
120 SCREEN 1
130 T = 1
140 SC = 0
150      '***** KRESLIT OBDELNIKOVY TERC A HRAT ***
160 CLS
170 GOSUB 1130  'KRESLIT TOULEC
180 J = 1        'J JE POSET POKUSU NA TENTO TERC
190 LOCATE 25,29
200 PRINT "SKORE";
210 LOCATE 25,35
220 PRINT SC;
230 XL = 100 + RND(1) * 180  'NAHODNE UMISTENI LEVEHO
     OKRAJE OBDELNIKA
240 XR = XL + 35
250 YT = RND(1) * 165       'NAHODNE UMISTENI HORNIHO
```

OKRAJE OBDELNIKA

```
260 YB = YT + 35
270 LINE(XL,YT)-(XL,YB)    'KRESLENI OBDELNIKA
280 LINE(XL,YB)-(XR,YB)
290 LINE(XR,YB)-(XR,YT)
300 LINE(XR,YT)-(XL,YT)
310 ***** VYBER UHLU A RYCHLOSTI SIPU ****
320 LOCATE 1,1
330 PRINT "
340 LOCATE 1,1
350 PRINT "UHEL (0-90)";
360 INPUT A
370 IF A >= 0 AND A <= 90 THEN 390
380 GOTO 350
390 A = A * 3.14159 / 180      'PREVO A NA RADIANY
400 LOCATE 1,19
410 PRINT "RYCHLOST";
420 INPUT S
430 G = 980                      'G JE GRAVITACNI SILA
440 LA = 40                       'LA JE DELKA SIPU
450 LT = 8                         'LT JE DELKA SPICKY SIPU
460   'URCENI ROZSAHU A VYSKY LETU SIPU
470 R = S * S * SIN(2 * A) / G   'R JE ROZSAH (NA OSE X)
     LETU SIPU
480   'URCENI KOEFICIENTU ROVNICE PARABOLY
490 C1 = G / (2 * (S * COS(A)) ^ 2)
500 C2 = - TAN(A)
510 GOSUB 1400                    'VYJMUTI SIPU Z TOLCE
520 ***** POHYB SIPU ****
530   'URCENI KONCOVEHO BODU SIPU NA PARABOLE A KRESLENI
540 FOR X = R/10 TO R STEP R/8   'UMISTENI SIPU NA SOUSEDNI
     DESETINU R
550   Y = C1 * X ^ 2 + C2 * X + YO
560   'X A Y JSOU SOURADNICE KONCOVEHO BODU NA PARABOLE
570   'URCENI DALSIHO KONCOVEHO BODU SIPU
580   M = C1 * X * 2 + C2          'M JE SMERNICE SIPU
590   A1 = ATN(M)                'INVERZNI TANGENTA DA UHEL A1
600   Y1 = Y + LA * SIN(A1)
610   X1 = X + LA * COS(A1)
620   IF X1 > XM OR Y1 > YM THEN GOSUB 830 'JE DRUHY
     KONCOVY BOD NA STINITKU ?
```

```

630 IF X > R/10 THEN GOSUB 1620      'VYMAZANI SIPU
640 'VYPOCET VRCHOLU SIPU
650 M2 = M + .75      'SMERNICE JEDNOHO VRCHOLU
660 A2 = ATN(M2)
670 X2 = X1 - LT * COS(A2)
680 Y2 = Y1 - LT * SIN(A2)
690 M3 = M - .75      'SMERNICE DRUHEHO VRCHOLU
700 A3 = ATN(M3)
710 X3 = X1 - LT * COS(A3)
720 Y3 = Y1 - LT * SIN(A3)
730 GOSUB 1560      'KRESLENI SIPU
740 IF X1 < XL OR X1 > XR THEN 780 'SIP NENI V OBDEL.
750 IF Y1 < YT OR Y1 > YB THEN 780
760      'SIP TREFIL OBDELNIK
770 GOTO 930
780 XS = X      'ULOZENI AKTUALNI POZICE DO XS,YS,X1S,Y1S
790 YS = Y
800 X1S = X1
810 Y1S = Y1
820 NEXT X
830 J = J + 1
840 IF J > 3 THEN 870  'POKRACOVAT DALSI TERCEM NEBO KONEC
850 GOSUB 1620
860 GOTO 310
870 IF T = 5 THEN 1710      'HRA SKONCILA
880 T = T + 1      'JINAK DALSI TERC
890 LOCATE 25,1
900 PRINT "PRILIS SPATNE. ZKUS DALSI";
910 FOR J = 1 TO 600: NEXT J
920 GOTO 160
930 ***** SIP TREFIL DO OBDELNIKA *****
940 'ZVYSENI SKORE
950 IF J = 1 THEN SC = SC + 10
960 IF J = 2 THEN SC = SC + 5
970 IF J = 3 THEN SC = SC + 2
980 LOCATE 25,35
990 PRINT SC;
1000 FOR TB = 1 TO 2      'BLIKANI "TREFA"
1010      LOCATE 25,1
1020      PRINT "TREFA!";
1030      FOR M = 1 TO 300: NEXT M

```

```

1040 LOCATE 25,1
1050 PRINT "      ";
1060 FOR M = 1 TO 300: NEXT M
1070 NEXT TB
1080 IF T <> 5 THEN 1100
1090 GOTO 1710
1100 T = T + 1
1110 GOTO 160           'POKRACOVAT NA DALSI TERC
1120
1130 ##### KRESLENI TOULCE #####
1140 LOCATE 22,1
1150 PRINT T;
1160 XC = 10: YC = 150: RX = 10: RY = 3
1170 FOR AQ = 0 TO 6.28318 STEP 1/10
1180   XQ = XC + RX * COS(AQ)
1190   YQ = YC + RY * SIN(AQ)
1200   PSET(XQ,YQ)
1210 NEXT AQ
1220 YC = 185
1230 FOR AQ = 0 TO 3.14159 STEP 1/10
1240   XQ = XC + RX * COS(AQ)
1250   YQ = YC + RY * SIN(AQ)
1260   PSET(XQ,YQ)
1270 NEXT AQ
1280 LINE(0,150)-(0,185)
1290 LINE(20,150)-(20,185)
1300 LINE(5,130)-(5,152)
1310 LINE(5,130)-(2,133)
1320 LINE(5,130)-(8,133)
1330 LINE(10,125)-(12,153)
1340 LINE(10,125)-(7,128)
1350 LINE(10,125)-(13,128)
1360 LINE(18,128)-(15,152)
1370 LINE(18,128)-(15,131)
1380 LINE(18,128)-(21,131)
1390 RETURN
1400 ##### VYJMUTI SIPU Z TOULCE #####
1410 REM
1420 IF J <> 1 THEN 1460
1430 LINE(10,125)-(12,153),0  'VYJMUTI PRVEHO SIPU
1440 LINE(10,125)-(7,128),0

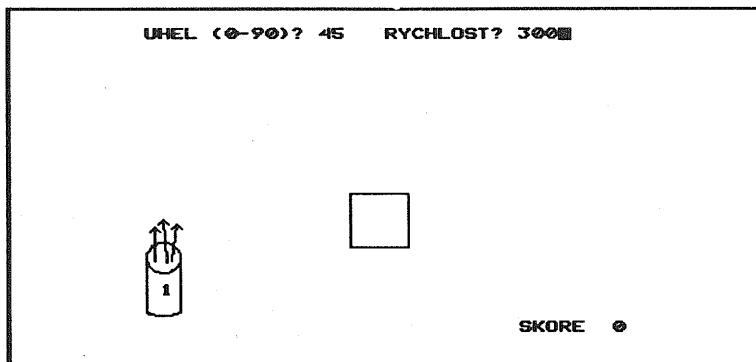
```

```

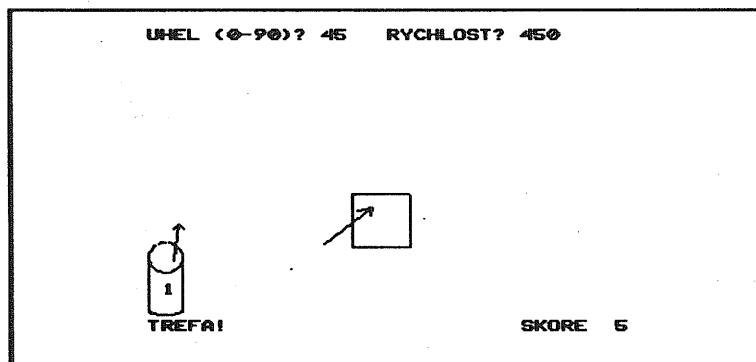
1450 LINE(10,125)-(13,128),0
1460 IF J <> 2 THEN 1500
1470 LINE(5,130)-(5,152),0      'VYJMUTI DRUHEHO SIPU
1480 LINE(5,130)-(2,133),0
1490 LINE(5,130)-(8,133),0
1500 IF J <> 3 THEN 1540
1510 LINE(18,128)-(15,152),0    'VYJMUTI TRETIHO SIPU
1520 LINE(18,128)-(15,131),0
1530 LINE(18,128)-(21,131),0
1540 REM
1550 RETURN
1560 '##### KRESLENI SIPU #####
1570 REM
1580 LINE(X,Y)-(X1,Y1),1
1590 LINE(X1,Y1)-(X2,Y2),1
1600 LINE(X1,Y1)-(X3,Y3),1
1610 RETURN
1620 '##### VYMAZANI SIPU #####
1630 REM
1640 LINE(XS,YS)-(X1S,Y1S),0
1650 LINE(X1S,Y1S)-(X2,Y2),0
1660 LINE(X1S,Y1S)-(X3,Y3),0
1670 REM
1680 RETURN
1690 '#####
1700 '
1710 ***** HRAT ZNOVA NEBO STOP *****
1720 CLS
1730 LOCATE 12,30
1740 PRINT "KONECNE SKORE -- "; SC
1750 PRINT
1760 PRINT TAB(10); "CHCES HRAT ZNOVA ?";
1770 INPUT CS
1780 IF CS = "N" THEN 1800
1790 GOTO 130
1800 END

```

Prog. 14-5 Hra šíp s terčem.



(a)



(b)

Obr. 14-5 Poloha šípu na začátku a při zásahu do černého pro lukostřeleckou hru (Prog. 14-5).