666. ZO SVAZARMU

TABULKOVÉ PROCESORY

KURS uživatelů osobních mikropočítačů

METODICKÝ MATERIÁL PRO UŽIVATELE MIKROPOČÍTAČŮ

Ing. Vít Libovický

Tabulkové procesory

učebnice
tabulkového procesoru

OMNICALC

pro
mikropočítač ZX Spectrum

Mikrokomputer Klub
P.S. 64, PRAHA 69

Dobrý den,

po prostudování uživatelské příručky k tabulkovému procesoru OMNICALC dnes přistupujete ke studiu učebnice. Vaše nedočkavost k ovládnutí světa tabulkových procesorů určitě způsobila, že při obdržení lekce listujete jak učebnicí, tak i příručkou. Mnozí z Vás po prvním ukojení zvědavosti, po letmém prolistování a nahrání programu OMNICALC do mikropočítače se pustí rovnou do uváděných příkladů. Zanechte tohoto bezkoncepčního přístupu. Nejprve prostudujte podle pokynů příručku a potom jste plně připraveni ovládnout i tuto učebnici.

V učebnici se setkáte s novým způsobem studia. Každá kapitola začíná tabulkou. V jednom z jejích políček najdete i položku POČET STUDIJNÍCH HODIN. Vyplňte ji pravdivě po prostudování každé lekce. Na závěr pak snadno budete moci usoudit zda doba, kterou jste vynaložili ke studiu odpovídá Vašim znalostem. Ani sebelépe zpracovaná učebnice totiž nemůže zaručit potřebnou úroveň Vašich znalostí, jestli že Vám k jejímu prostudování stačilo pár hodin.

Jsem přesvědčen, že Vy, kteří se diferencujete již svým vlastním zájmem o věc, přistoupíte ke studiu zodpovědně. A tak i na osmibitové hračce Spectrum pomocí Dálkového kursu uživatelů osobních mikropočítačů se z Vás stanou profesionálně připravení uživatelé malé výpočetní techniky. A nejen to. Budete tuto techniku, na rozdíl od řady profesionálů, umět ve svém současném či budoucím oboru využívat. Tak Vás určitě nezaskočí až se jednou i Vaše Spectrum promění na mikropočítač s operačním systémem CP/M či rovnou na IBM PC.

Štefan Kratochvíl

Štefan Kratochvíl
Středisko pro mládež a elektroniku
Centra pro mládež, vědu a techniku ÚV SSM

Obsah

1. Úvod 8

2. Co je k použití OMNICALCu potřeba 10

3. Základní seznámení s možnostmi OMNICALCu 14
3.1. Proč "tabulkový procesor" 14
3.2. Jednoduchý problém: finanční plánování 15
3.3. Síla tabulkového procesoru 17
3.4. Co se stane když... ? 18
3.5. Použití elektronické tabulky 19
3.6. Vestavěné funkce 20
3.7. Prázdné formuláře - vzory 20
3.8. Další možnosti OMNICALCu 21
3.9. Závěrečný přehled 22

4. Nadpisy, čísla a vzorce 24
4.1. Nadpisy 24
4.2. Vyhodnocování textů použitých ve výrazech . . 25
4.3. Číselné konstanty 27
4.4. Zobrazení čísel kontra uložení čísel 27
4.5. Vzorce 28
4.6. Časová náročnost výpočtů 30
4.7. Součet 32
4.8. Cyklické odkazy 34
4.9. Dopředné odkazy 36

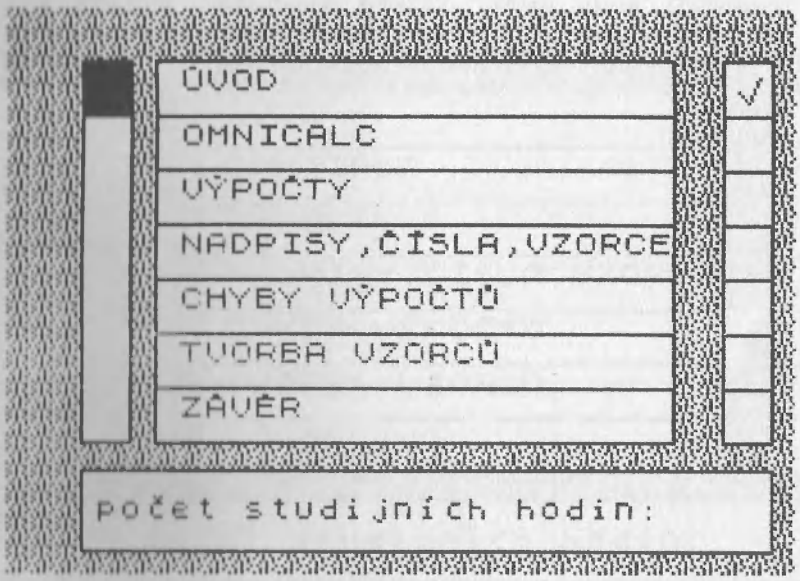
5. Rozpoznávání, prevence a oprava chyb 42
5.1. Vlastní prostředky OMNICALCu k předcházení chybám 42
5.2. Základní chyby 43
5.3. Ještě několik typických chyb 47
5.4. Detekce a prevence chyb 48
5.5. Oprava chyb 52

Obsah

5.6. Shrnutí 53

6. Tvorba vzorů 55
6.1. Analýza problému 56
6.2. Symbolika pro identifikaci typu políček . . . 58
6.3. Vložení dokumentace do formuláře 59

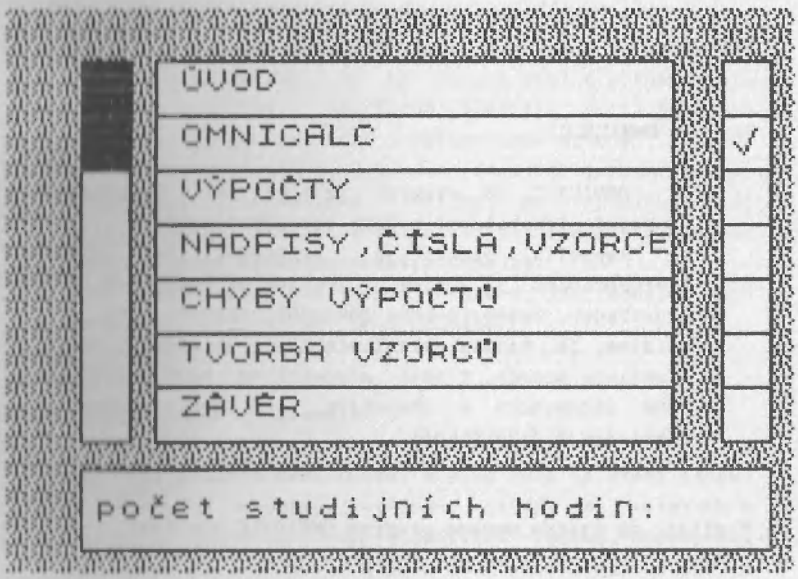
7. Závěr 62


ÚVOD
OMNICALC
VÝPOČTY
NADPISY,ČÍSLA,VZORCE
CHYBY VÝPOČTŮ
TVORBA VZORCŮ
ZÁVĚR
√
počet studijních hodin:

1. Úvod

Dostává se vám do ruky učebnice práce s tabulkovým procesorem - **spreadsheetem**. Vzhledem k tomu, že nejrozšířenější mikropočítač v Československu je **Sinclair ZX-Spectrum**, byl zvolen program **OMNICALC**, který pracuje právě na tomto typu mikropočítače. Jedná se sice o dosti jednoduchý program z této oblasti programů, ale pro účely vysvětlení principů práce s podobnými programy svými možnostmi postačuje.

Tato kniha není uživatelskou příručkou k ovládání programu OMNICALC a nepopisuje detaily jeho ovládání. Doporučuji proto napřed důkladně prostudovat příručku k OMNICALCu a vyzkoušet si jeho ovládání v praxi na počítači.


ÚVOD
OMNICALC
VÝPOČTY
NADPISY,ČÍSLA,VZORCE
CHYBY VÝPOČTŮ
TVORBA VZORCŮ
ZÁVĚR
počet studijních hodin:

2. Co je k použití OMNICALCu potřeba

Používání OMNICALCu je nejlepší cestou k pochopení problematiky programů typu tabulkového procesoru. K práci s ním nestačí pouze program OMNICALC jako takový, ale je nutný celý systém objektů:

- program OMNICALC
- počítač, na kterém budeme program používat
- problém, který chceme řešit
- data, která budeme zpracovávat
- osoba, která bude tento systém používat

V této kapitole stručně rozebereme uvedené prvky tohoto systému.

Program OMNICALC

OMNICALC je program z oblasti programů pro obchodní, laboratorní a jiné tabulkové výpočty. Vzhledem ke kapacitě paměti a rychlosti 8mi bitových mikropočítačů se jedná o velice jednoduchý a omezený prostředek. Ovšem všechny základní funkce plní, a jak uvidíme, je bohatá oblast problémů, se kterými nám může efektivně pomoci. V této učebnici se bude pojednávat o jeho možnostech a omezeních, jaké problémy pomocí něj řešit lze a jaké nikoli.

Počítač, na kterém budeme program OMNICALC používat

Program OMNICALC je určen pro mikropočítač Sinclair ZX-Spectrum 48K a jeho novější modifikace. Jedná se o

nejrozšířenější mikropočítač v našem státě a předpokládá se, že čtenář této učebnice ho vlastní nebo má k němu prístup.

Omezení způsobená typem použitého počítače u této kategorie programů nejsou tak podstatná. Příkladem může být program VisiCalc, který pracuje na mnoha typech velice odlišných počítačů (např. Hewlett Packard 85 - Atari 800 - TRS-80) a jeho ovládání i možnosti jsou na všech typech počítačů naprosto shodné.

Problém, který chceme řešit

Uvidíme, že pro použití OMNICALCu pro řešení problémů budeme vytvářet tzv. **elektronické formuláře** (electronic sheet). To se budeme učit v průběhu této učebnice. Nejdříve je nutné vytvořit popis řešeného problému (struktura dat, matematické vztahy), dále na základě tohoto popisu vytvořit formulář (zapsat texty, konstanty, matematické vztahy do tabulky), do připraveného formuláře zadávat konkrétní údaje a nakonec takto získané výsledky efektivně použít.

V samostatné kapitole si ukážeme, jak rozpoznávat a opravovat chyby a jak jim předcházet.

Data, která budeme zpracovávat

Při práci s OMNICALCem, stejně jako se všemi jinými počítačovými systémy, výsledky závisí na vztazích a vstupních hodnotách, které počítači zadáme. Můžeme mít provedenu přesnou analýzu problému, můžeme mít připraven perfektní formulář, ale když zadáme chybné údaje, museli

Co je k použití OMNICALCu potřeba

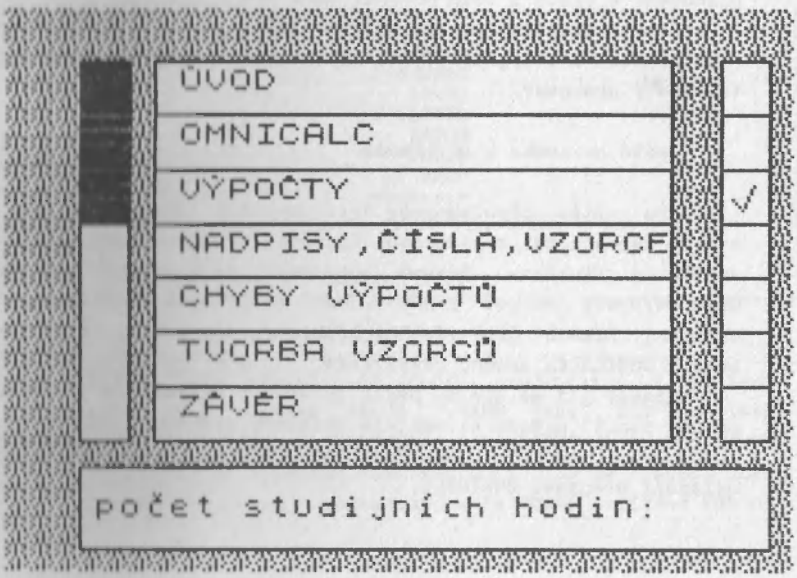
by jsme mít opravdu z pekla štěstí, aby výsledky byly správné.

Uvidíme, že OMNICALC má velice omezené možnosti pro kontrolu správnosti dat a že je tedy nutné zadávání údajů věnovat velkou pozornost.

Osoba, která bude tento systém používat

Pokud chceme používat OMNICALC efektivně, musíme být plně informování o jeho možnostech. Je také možné, aby jiné osoby používaly vámi připravené formuláře. V tomto prípadě musíte dbát o to, aby i tyto osoby měly dostatečné znalosti o používání tohoto systému.

V textu knihy bude uvedeno množství aplikací OMNICALCu, tak aby jste získali přehled o oblasti problémů, které můžete řešit. Složitost těchto příkladů by měla od velice jednoduchých postupně gradovat ke složitějším.


ÚVOD
OMNICALC
VÝPOČTY
NADPISY,ČÍSLA,VZORCE
CHYBY VÝPOČTŮ
TVORBA VZORCŮ
ZÁVĚR
✓
počet studijních hodin:

3. Základní seznámení s možnostmi OMNICALCu

3.1. Proč "tabulkový procesor"

OMNICALC je pružný a efektivní pracovní prostředek. Jeho síla spočívá ve schopnosti provádět za nás operace, na které bychom potřebovali **tužku**, **papír** a **kalkulačku**. Kombinace rychlosti počítače a inteligence programového vybavení nám dává prostředky k rychlému a pružnému řešení problémů. Proto tento systém nalézá mnohostranné uplatnění v praxi.

Dvěmi základními oblastmi použití jsou **účetnictví** a **plánování**. Tyto dvě oblasti lze spojit do **finančního plánování** a právě z této oblasti bude první příklad.

Řešené problémy se zapisují do tabulky, která má jakousi "vestavěnou inteligenci". Odsud také plyne pojmenování **tabulkový procesor**.

Ještě poznámku k příkladům.

Pro větší přehlednost byla zvolena grafická úprava, která neodpovídá skutečnému formátu na displeji nebo na výpisu z tiskárny. Budeme ignorovat, že v OMNICALCu nelze psát poznámky malými písmeny ani s českými diakritickými znaménky. Přesto jsou všechny příklady voleny tak, že je lze pomocí OMNICALCu snadno realizovat.

Čtenář již má jistou praxi ze studia příručky a jistě mu nebude činit potíže si některé příklady prakticky odzkoušet. Tuto praxi naopak doporučuji pro hlubší zapamatování si ovládání programu OMNICALC.

3.2. Jednoduchý problém: finanční plánování

Naši sondu do finančního plánování začneme s informací o účtu malého podniku nebo spíše družstva. Počáteční informace vidíme na obrázku 3-1. Samozřejmě se jedná o jednoduchý příklad, který nemá s praxí mnoho společného, ale je vhodný pro úvod do problematiky. Na obrázku 3-1 vidíme několik řádek s čísly a ve spodu jeden sloupcový součet. Ale i na této primitivní úrovni si můžeme demonstrovat problémy plánování.

Plán

	ROK			
	0	1	2	3
Zaměstnanců:	50			
Platy	1800000			
Daně	450000			
Telefon	10000			
Odvod	100000			
Cestovné	10000			
Reklama	20000			
Různé	5000			
	2395000			

Obrázek 3-1.
Stav výdajů v základním roce.

PLATY jsou závislé na počtu zaměstnanců a počítáme průměrný roční příjem 36000, **DANĚ** jsou 25% z platů zaměstnanců, platby za **TELEFON** počítáme 200 na osobu, **ODVOD** je konstantní, na **CESTOVNÉ** počítáme opět 200 na osobu, účet za **REKLAMU** je také konstantní a na **RŮZNÉ** počítáme 100 na osobu.

Celkové sumy pro 50 zaměstnanců vidíme na obrázku 3-1. Nyní začneme trochu plánovat. Budeme uvažovat, že každý rok

přijmeme 5 zaměstnanců. Tím nám položky závislé na počtu zaměstnanců úměrně vzrostou.

Nyní ale musíme začít počítat s růstem platů i nákladů. takže PLATY, TELEFON, REKLAMA a RŮZNÉ budou mít nárůst o 10% každý rok, CESTOVNÍ VÝDAJE 20% a pouze ODVOD zůstává konstantní.

Zkuste vzít do ruky kalkulačku a spočítat položky na tři roky dopředu! Pokud navíc vaše kalkulačka neumí mocnit desetinným číslem, máte na hodnou chvíli co dělat.

Pokud jste neudělali chybu, dostali jste výsledky jako na obrázku 3-2.

Plán

	ROK 0	1	2	3
Zaměstnanců:	50	55	60	65
Platy	1800000	1980000	2178000	2395800
Daně	450000	495000	544500	598950
Telefon	10000	11000	12100	13310
Odvod	100000	100000	100000	100000
Cestovné výdaje	10000	12000	14400	17280
Reklama	20000	22000	24200	26620
Různé	5000	5500	6050	6655
	2395000	2625500	2879250	3158615

Obrázek 3-2.
Stav výdajů v následujících třech letech.

Úspěšně jsme zvládli úskalí numerické matematiky a máme radost ze správných výsledků. Co když ale budeme předpokládat místo 5 přírůstek 6 zaměstnanců za rok! Ruční výpočet představuje vzít čistý papír, na něj přenést výchozí údaje (i obyčejné opisování údajů představuje vážné nebezpečí vzniku chyby) a znovu spočítat všechny položky závislé na počtu zaměstnanců, což jsou skoro všechny.

Právě v tomto okamžiku oceníme práci s elektronickou tabulkou, vždyť právě na takovéto úkony je vymyšlena!

3.3. Síla tabulkového procesoru

OMNICALC je ideální systém pro řešení našeho plánovacího problému. Snadno a rychle vytvoříme formulář obsahující pojmenování řádek a sloupců, aktuální hodnoty pro počáteční rok a vzorce pro výpočet všech ostatních žádaných údajů podle našich předpokladů.

Jak vypadá řešení našeho problému pomocí OMNICALCu?

Formulář řešící náš problém zadáme do elektronické tabulky následujícím způsobem:

1. do řádek **g-m** zapíšeme do prvního sloupce název položky a do druhého sloupce vzorce (viz dále)
2. do řádky **o** vložíme vzorec pro součet řádků **g-m**
3. připravený druhý sloupec se vzorci "rozmnožíme" trojnásobným vložením sloupce (kurzor je nastaven na druhý sloupec)
4. řádky a-f poslouží pro zápis textové hlavičky
5. do řádky c zapíšeme čísla roků 0..3
6. do řádky **e** vložíme počty zaměstnanců v jednotlivých letech

Pro jednotlivé položky zadáme do druhého sloupce tyto vzorce:

Platy:	36000xe2x1.1↑c2
Daně:	g2x0.25
Telefon:	200xe2x1.1↑c2
Odvod:	100000
Cestovní v.:	200xe2x1.2↑c2
Reklama:	20000x1.1↑c2
Různé:	100xe2x1.1↑c2
součet:	g2+h2+i2+j2+k2+l2+m2

Obrázek 3-3.
Zadání položek v OMNICALCu

Formulář máme připraven, vložíme tedy do řádky **e** předpokládané počty zaměstnanců: 50, 56, 62, 68.

Nakonec necháme formulář spočítat a za 4 sekundy máme správné výsledky!

3.4. Co se stane když... ?

Formulář včetně všech dat je připraven. Již víme, jak se změní výsledky, když přijmeme každý rok 6 lidí místo 5, ale jak se projeví, když budeme méně telefonovat nebo když budeme více cestovat nebo když se sníží daně? Je velice jednoduché změnit jeden vzorec podle našeho nápadu a přesvědčit se o výsledku této akce. Když je nutné vše počítat na kalkulačce, kdo by si přidělával práci, ovšem když to za nás spočítá počítač, to již stojí za to! Na naše otázky a nápady dostáváme snadno a rychle přesné výsledky. Za stejný čas máme k dispozici podstatně více údajů, podle kterých můžeme **rozhodovat** a **řídit** běh podniku. To je hlavní síla těchto systémů, v našem případě OMNICALCu.

I z tohoto jednoduchého příkladu je zřejmý přínos počítačových systémů, vždyť největší problém našeho národního hospodářství leží právě v oblasti hospodářského řízení.

3.5. Použití elektronické tabulky

Program OMNICALC představuje tabulku. Příprava formuláře pro řešený problém je vlastně vyplňováním tabulky. Připomeňme si, že řádky této tabulky jsou značeny písmeny abecedy **a-z** a dále dvoupísmennými kombinacemi **aa-az**, **ba-bz** atd. Sloupce jsou značeny čísly od 1 do 99. Na políčka této tabulky se odvoláváme souřadnicemi, např. a1, g35 nebo ab29. Do každého z těchto políček lze vepsat údaj. Je nutné rozlišovat typy údajů, které jsou tyto:

- text
- číselná hodnota
- výraz
- součet
- prázdné políčko

Podle zobrazení na displeji nebývá zřejmé, o políčko jakého typu se jedná. Například se nám v políčku zobrazuje toto:

156

Zdá se, že se jedná o číselnou hodnotu. Ale nemusí tomu tak být. Toto políčko může být **libovolného** typu!

Problém spočívá v tom, že v závislosti na typu tohoto políčka můžeme dostávat různé výsledky. Podrobněji bude o této problematice pojednáno v kapitole o chybách.

3.6. Vestavěné funkce

Ve výrazech je možné použít libovolné funkce BASICu - základních aritmetických, goniometrických, logických a dalších funkcí. V účetnických aplikacích si většinou vystačíme s několika základními operacemi (+, -, *, /, - sčítání, odčítání, násobení, dělení a mocnění), ale jistě si sami vymyslíte aplikace, kde použijete i další matematické funkce.

Zvláštností OMNICALCu oproti jiným systémům je speciální typ políčka pro součet. Standardně bývá k dispozici funkce **SUM**, která se použije ve výrazu stejně jako všechny jiné funkce.

3.7. Prázdné formuláře - vzory

Jedním z možných přístupů k používání OMNICALCu je oddělení tvorby formuláře od konkrétního použití pro výpočet.

Podívejte se na obrázek 3-4. Je to obdobný formulář, jako jsme již viděli, ale v mnoha políčcích jsou pouze tečky místo čísel a místo dalších výsledků jsou nuly.

```
                    Pětiletý plán založený na
          ....       % zvyšování výdajů ročně

oblast                   rok      rok      rok
                           0        1        2
                       =====================
výzkum                  ....        0        0
vývoj                   ....        0        0
výroba                  ....        0        0
                       =====================
celkem:                    0        0        0
```

Obrázek 3-4.
Prázdný formulář - vzor.

Tento obrázek je ukázkou připraveného formuláře - vzoru. Uživatel vyplní pouze políčka s tečkami svými číselnými údaji a dostane výsledek. Pro jiné číselné údaje použije znovu připravený prázdný formulář.

Výhodou tohoto přístupu je, že formulář připraví osoba s dobrou znalostí ovládání použitého programu, zatímco další práci zápisu číselných hodnot lze svěřit i neodborníkovi po krátkém zaučení (například sekretářce). Navíc stejný formulář lze používat na dalších pracovištích, kde si ho již nemusí připravovat. To může vést ke značné úspoře práce, času a papíru.

3.8. Další možnosti OMNICALCu

Zde ve zkratce shrneme základní možnosti práce s OMNICALCem.

- Snadné vkládání a rušení řádků a sloupců.
- Dva módy zobrazení čísel - zaokrouhleně na celá čísla nebo na dvě desetinná místa.

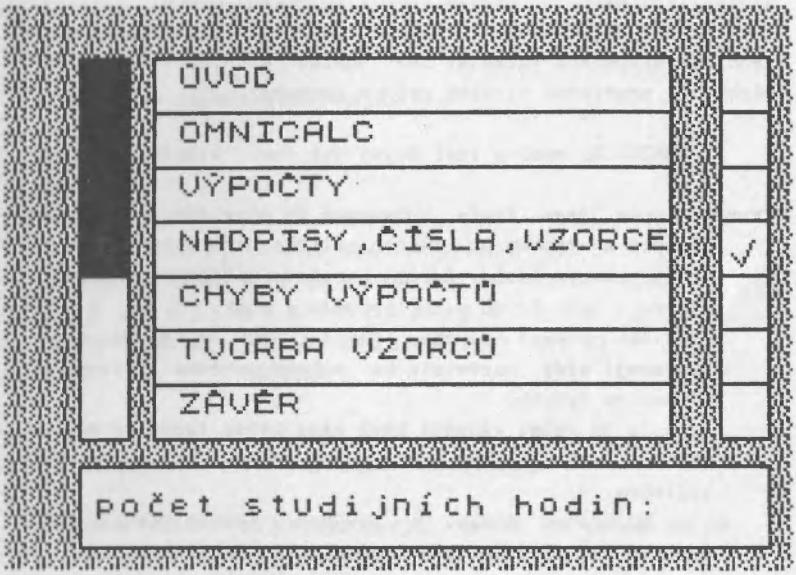
- Celý formulář nebo jeho část může být vytištěn na tiskárně.
- Zadané nebo vypočtené hodnoty lze zobrazit a vytisknout ve formě sloupcového grafu.
- Formuláře nebo číselná dat lze zaznamenávat na magnetofonové kazetě. Z připraveného formuláře lze vytvořit libovolné množství kopií.

3.9. Závěrečný přehled

V této kapitole jste se dočetli o základních možnostech OMNICALCu, principech jeho použití a výhodách proti ručnímu způsobu práce:

- elektronická tabulka, do které můžeme vepisovat texty, číselné hodnoty a vzorce
- schopnost přesného a rychlého výpočtu ihned po zadání nových hodnot
- možnost využití všech matematických funkcí BASICu pro snazší zápis výrazů
- možnost tvorby prázdných formulářů - vzorů
- funkce pro globální změny v tabulce (vkládání a rušení řádků a sloupců)
- dva možné módy zobrazení čísel
- funkce pro tisk celého formuláře nebo jeho částí
- možnost záznamu a archivace formulářů včetně číselných dat

V následujících kapitolách této učebnice získáte podrobné znalosti pro použití těchto možností a informace o omezeních OMNICALCu.


ÚVOD
OMNICALC
VÝPOČTY
NADPISY,ČÍSLA,VZORCE
CHYBY VÝPOČTŮ
TVORBA VZORCŮ
ZÁVĚR
počet studijních hodin:

4. Nadpisy, čísla a vzorce

V této kapitole získáte podrobné informace o zápisu používání nadpisů, čísel a vzorců. Také se dovíte o zákazech a omezeních. Začneme s nadpisy.

4.1. Nadpisy

Při práci s elektronickým formulářem nás zajímají hlavně číselné výsledky. Na tyto výsledky nemají žádné nadpisy vliv. Ovšem sloupce čísel bez popisu jsou velice nepřehledné a při jejich zápisu snadno můžeme ztratit přehled o tom, co zrovna píšeme, což by bylo zdrojem chyb a zdržování práce.

Pokud mají naše formuláře používat další osoby, je sebelépe vymyšlený formulář bez popisu k ničemu. Z těchto důvodů je používání titulků velice žádoucí.

V OMNICALCu musíme rozlišovat dva druhy titulků:

Prvním typem jsou texty vpisované do hlavičkových políček sloupců a řádek. Do těchto políček lze vpisovat pouze texty, nikoli čísla. Oproti ostatním políčkům mají malou výhodu v tom, že do nich lze psát i znaky (,), |, | a ˇ. Ovšem hlavní výhodou je to, že se **na displeji zobrazují vždy**, nezávisle na poloze okénka, kterým se díváme na tabulku.

To je velmi výhodné když máme větší formulář než je okénko 3x15 políček, což je spíše pravidlem než vyjímkou.

Na druhou stranu je nevýhodou těchto hlavičkových políček to, že jsou jenom jedny zcela nahoře a vpravo.

Speciální funkcí lze do hlavičkových políček vepsat zkratky měsíců roku. Tato funkce je užitečná pro

usnadnění práce například při výpočtu měsíčních uzávěrek.

Druhým typem titulků jsou texty vpisované do normálních políček tabulky. Do jednoho políčka lze vepsat text o maximální délce 7 znaků a lze použít pouze velká písmena, číslice a některé další grafické znaky. Pokud napíšeme delší text, ořízne se zprava. Text kratší než oněch sedm znaků se do políčka vypisuje zleva. Pokud chceme mít text zarovnaný doprava, musíme před něj vepsat příslušný počet mezer.

Delší nadpis než sedm znaků musíme rozepsat do několika políček vedle sebe. Přitom ovšem mezi těmito políčky zůstane jednoznaková mezera - světle modrý sloupeček.

Uvečme několik příkladů textů:

```
CELKEM:
KČS
(xxxxx)
======= ======= ======= ======= =======
MESICNI UZAVERKA ZA ROK 1987
M E S I C N I  U Z A V E R K A  Z A  R O K  1 9 8 7
LDN`87  UNR`87  BZN`87  DBN`87  KTN`87  CVN`87
1987    1       0000000
```

4.2. Vyhodnocování textů použitých ve výrazech

Všechny texty se vyhodnocují, pokud se odkaz na ně objeví ve výrazu, jako hodnota nula. To může být někdy výhoda a jindy nevýhoda. Výhodné je to při použití součtů. Můžeme klidně podtrhnout sloupec čísel textem

`-------` nebo `=======`

bez nebezpečí chyby v součtu, protože +0 výsledek nezmění.

Nevýhodné může být vyhodnocování textů jako 0 v případech, kdy počítáme rozdíl dvou sloupců a v těchto sloupcích použijeme textu pro podtržení. Vzorce pro rozdíl získáme opakováním jednoho vzorce do celého sloupečku. Výsledek rozdílu dvou textů se objeví jako 0.

Tento případ je ukázán na obrázku 4-1. Do sloupečků 1 a 2 jsme napřed zapsali několik hodnot, podtrhli a zadali součtové políčko dospodu. Potom teprve jsme do sloupce 3 zapsali vzorec pro rozdíl

a1-a2

a pomocí opakování až k políčku **e3** jsme získali sloupec rozdílů. V políčku **d3** se zobrazí 0, nikoli podtržení, jak by jsme chtěli. Tento kaz na kráse musíme odstranit dodatečným zapsáním podtržení do tohoto políčka.

	1	2	3
a	20	15	5
b	5	4	1
c	8	10	-2
d	---	---	0
e	33	29	4

Obrázek 4-1.
Příklad se vzorcem pro rozdíl dvou textových políček.

4.3. Číselné konstanty

Číselné konstanty se v OMNICALCu neliší od čísel používaných v BASICu. Skládají se z číslic **0** až **9**, mohou obsahovat desetinnou tečku a mohou začínat znaménkem **+** nebo **-**. Žádný jiný znak (čárka, $ apod.) nesmí být součástí čísla.

správná čísla	chybná čísla
1000	1,100
-247.56	$12.13
+197.000	42.00KCS
0002	DESET

OMNICALC ignoruje nuly zleva a nuly zprava za desetinnou tečkou.

4.4. Zobrazení čísel kontra uložení čísel

OMNICALC počítá se stejnou přesností jako BASIC, tedy na 8 platných míst. Podle zvoleného módu se čísla zobrazují jako celá (mód 1) nebo na dvě desetinná místa (mód 2). Čísla se nezaokrouhlují, ale ořezávají! Výpočet ovšem vždy probíhá s plnou přesností 8 platných míst.

napsali jsme	mód 1	mód 2	v paměti je
1	1	1.00	1
75	75	75.00	75
.0173	0	0.01	0.0173
745.896	745	745.89	745.896
1000000	1000000	1000000.00	1000000
100000000	1E+8	1E+8	100000000
1000000.01	1000000	1000000.00	1000000
100000000/3	33333333	33333333.00	33333333
SIN 3	0	0.14	.14112001

Z uvedených příkladů je vidět, že jediné nebezpečí chyby plyne z překročení 8 platných míst. Zdálo by se, že 8 míst bohatě stačí, ovšem opak je pravdou. Mějme účetnictví počítající na haléře a maximální přesné číslo je pouze jeden milión! Na toto omezení je třeba myslet při analýze aplikací.

Také je potřeba počítat s tím, že čísla jsou zobrazována jinak než se s nimi počítá!

4.5. Vzorce

Vzorce se používají pro zápis matematických vztahů mezi políčky formuláře. Vzorce jsou matematické výrazy, tak jak je známe z programovacích jazyků. Součastí vzorců mohou být číselné konstanty, souřadnice políček nebo vestavěné funkce.

vzorec	komentář
156	I pouhá konstanta může představovat vzorec.
18000+21000	Toto je jednoduchý vzorec pro součet dvou číselných konstant. Můžeme napsat takovýto vzorec, pokud používáme OMNICALC jako kalkulátor.
12+b3	Tento vzorec přičte 12 k hodnotě políčka b3.
b3+12	Samozřejmě totéž jako předchozí vzorec, neboť i v OMNICALCu platí komutativnost.
(b3+12)	Opět matematicky totéž. Závorky v tomto případě nemají význam.
230+2*PI	Použití vestavěné konstanty.
a1*b1|3	Tento vzorec demonstruje použití znaku pro mocnění (↑).

V OMNICALCu máme k dispozici tři logické, šest porovnávacích a pět matematických operátorů. Jsou to

OR	logický součet
AND	logický součin
NOT	logická negace
=	rovno
<	menší než
<	menší než
>=	větší nebo rovno
<=	menší nebo rovno
<>	nerovno
+	sčítání
-	odčítání
/	dělení
*	násobení
↑	mocnění

Tím se dostáváme k prioritě těchto operací, tedy k pořadí, v jakém se budou jednotlivé operace provádět, pokud není závorkami toto pořadí učeno. I v OMNICALCu je priorita operací standardní a odpovídá pořadí v této tabulce, s tím že mocnina má prioritu největší (provede se první), sčítání s odčítáním a násobení s dělením mají prioritu shodnou stejně jako všechny porovnávací operátory.

Výsledkem porovnání je **0** (nula), když srovnání neplatí a **1** (jedna) pokud porovnání platí.

Logické operátory se smí aplikovat i na čísla. Ovšem výsledky těchto operací jsou trochu neobvyklé a nejlépe si je ukážeme na příkladech.

10>8	**1**	platí
10<8	**0**	neplatí
0>2>1	**0**	vyhodnocuje se zleva
10 AND 5	**10**	x AND y je x, když y <> 0
10 AND 0	**0**	x AND y je 0, když y=0
10 AND -13	**10**	AND není komutativní!
-13 AND 10	**-13**	
12 OR 7	**1**	x OR y je 1 pokud x nebo y není 0
0 OR 0	**0**	0 OR 0 je 0
NOT 17	**0**	NOT x je 1 pro x <> 0
NOT -255	**0**	
NOT 0	**1**	NOT 0 je 1

4.6. Časová náročnost výpočtů

I když většina formulářů vyžaduje pro výpočet několik málo sekund nebo ještě méně, některé formuláře mohou vyžadovat pro výpočet mnohem více času.

Vezměme si například tento vzorec

AX↑3+BX↑2+CX+D.

Inicializujeme tabulku na rozměr 3x240. Konstanty A, B, C a D zvolíme 12, 6, 3 a 5. Proměnnou X umístíme do políčka a1. Do políčka c1 vložíme tento vzorec:

12*ka1^3+6*ka1^2+3*ka1+5

Opakováním tento vzorec umístíme do celého sloupce až do políčka ib1.

Do políčka a1 vložíme číslo 1 a necháme formulář spočítat. Výpočet trvá skoro celou minutu!

Nyní stejným způsobem zadáme vzorec:

((((ka1*12)+6)*ka1)+3)*ka1)+5

Opět spustíme výpočet. Tentokrát stačí čas do 8 sekund!

Účelem tohoto příkladu bylo ukázat, že čas potřebný pro výpočet formuláře může být až několik minut, ale hlavně že je silně závislý na tom, jak příslušné vzorce zadáme.

Ve formulářích, kde se používají základní aritmetické funkce (+, -, *, /) nebývá čas výpočtu problémem. Ale při použití vestavěných funkcí jako je mocnění, logaritmy, goniometrické funkce může čas činit problémy. Nejnepříjemnější je to u formulářů, které se používají často a které se často nechávají počítat.

Zde uvedeme pár zásad, kterými je možné výpočet urychlit.

pokud jedna počítaná hodnota je používána na více místech, je vhodné ji vypočítat jednou pomocí pomocného políčka a v ostatních výrazech se odvolávat na toto pole.

- trigonometrické funkce jsou obzvlášť pomalé. Předchozí zásada se zde vyplatí i pro konstanty typu SIN PI/2 apod.
- i jiné počítané konstanty spočítat pouze jednou (např. 2*PI, 3/2 apod.)
- pokud je to možné, použijte radši násobení než mocnění

Někdy se vyplatí vyzkoušet několik možností zápisu vzorců a vybrat si ten nejrychlejší.

4.7. Součet

Pro výpočet součtu řádky nebo sloupce slouží speciální typ políčka - **součtové pole**.

V OMNICALCu máme k dispozici dva typy součtu - **celkový součet** a **podsoučet**.

Podsoučtové políčko je určeno pro výpočet mezivýsledků - částečných součtů skupin dat. Součtové políčko slouží k výpočtu součtu všech hodnot ve sloupci nebo řádku.

Použití obou typů součtů si ukážeme na příkladu, který je zobrazen na obrázku 4-2. Jedná se o výpočet výdajů za různá časová období - týden, měsíc, čtvrtletí a rok.

	1	2	3	4	
a	Rok:	`87			
b					
c	Leden	Duben	Červ.	Říjen	
d	440	853	993	646	
e	430	836	480	919	
f	859	737	467	994	
g	627	503	684	550	
h	-------	-------	-------	-------	
i	2356	2929	2624	3109	
j					
k	Únor	Květen	Srpen	Listopd	
l	639	990	739	869	
m	965	892	694	779	
n	783	705	906	629	
o	994	545	617	807	
p	-------	-------	-------	-------	
q	3381	3132	2956	3084	
r					
s	Březen	Červen	Září	Prosinc	
t	552	547	520	982	
u	475	678	447	852	
v	628	862	932	761	
w	726	903	755	524	
x	-------	-------	-------	-------	
y	2381	2990	2654	3119	
z					
aa	==========	==========	==========	==========	CELKEM:
ab	8118	9051	8234	9312	34715

Obrázek 4-2.
Příklad použití součtů a podsoučtů.

V řádcích **i**, **q** a **y** jsou **sloupcové podsoučty** - součet za měsíc, v řádku **ab** jsou **sloupcové součty** - součty za čtvrtletí a v políčku **ab5** je **řádkový součet** - celkový součet za celý rok.

4.8. Cyklické odkazy

V OMNICALCu je možné zapsat do políčka vzorec, který se odkazuje opět na toto vlastní políčko. Například do políčka **a1** můžeme zapsat vzorec

a1+1.

Takovémuto odkazu sama na sebe se říká **cyklický odkaz**. Ukažme si jeho funkci na příkladu.

Po inicializaci tabulky vložíme do políčka **a1** uvedený vzorec. Na displeji se objeví v tomto políčku číslo 1. Prázdné políčko má totiž hodnotu 0. Naše políčko bylo v okamžiku vkládání vzorce prázdné, mělo tedy také hodnotu 0. Když k nule přičteme 1, dostaneme tento výsledek.

Nechte nyní tabulku několikrát za sebou přepočítat. V políčku **a1** se objevují čísla 2, 3, 4, 5 atd. Je vidět, že výsledek závisí na počtu přepočítání tabulky. To je velice nepřehledné a nepředvídatelné chování. Snažme se proto takovéto cyklické odkazy nezadávat.

Ovšem je také možné tohoto efektu využít. Řekněme, že nás zajímá úrok za jeden rok v závislosti na základním vkladu a úrokové míře.

Připravíme si formulář takto: do políčka **a2** vložíme krok zvyšování úrokové míry 0.01 (po 1%), řádek b zatím vynecháme, do políčka **e1** vložíme vzorec

d1+100,

který zopakujeme do dalších řádků prvního sloupce a nakonec do políčka **e2** vložíme vzorec

kb2xe1.

Po přepočítání vypadá formulář jako na obrázku 4-3. Nyní do políčka b2 vložíme vzorec s cyklickým odkazem

a2+b2.

Po přepočtení se v políčku b2 objeví hodnota 0.02 a ve druhém sloupci dostaneme požadované výsledky. Po každém dalším přepočtení se zvýší hodnota políčka **b2** o 0.01 a ve druhém sloupci budou nabíhat příslušné výsledky.

	1	2	3
a	Krok:	0.01	
b	%:		
c			
d	vklad	úrok	
e	100.00	0.00	
f	200.00	0.00	
g	300.00	0.00	
h	400.00	0.00	
i	500.00	0.00	
j	600.00	0.00	
k	700.00	0.00	
l	800.00	0.00	
m	900.00	0.00	
n	1000.00	0.00	
o	1100.00	0.00	

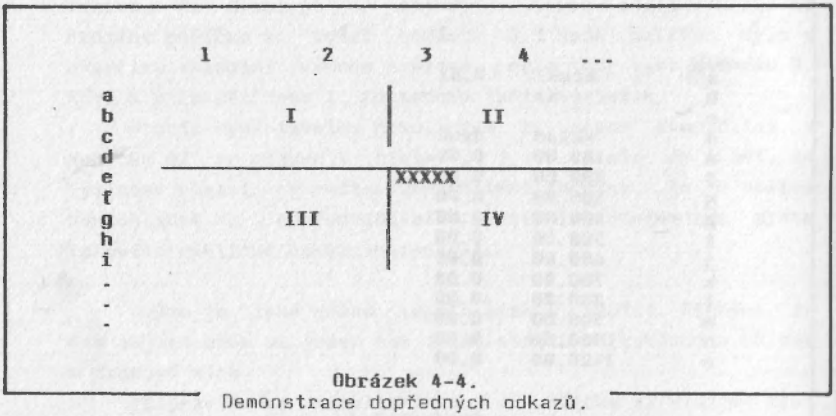
Obrázek 4-3.
Připravený formulář před zápisem vzorce s cyklickým odkazem.

Z príkladu jsme viděli, že použití cyklického odkazu může mít i praktické uplatnění, ale při jeho použití je nutné, abychom přesně věděli, co děláme, jinak dostaneme neočekávané výsledky.

Navíc pro vložení počáteční hodnoty, kterou budeme cyklicky zvyšovat pomocí cyklického odkazu, musíme do příslušného políčka nejprve vložit tuto hodnotu a potom znovu vložit celý vzorec.

4.9. Dopředné odkazy

O **dopředném odkazu** hovoříme tehdy, jestliže ve vzorci použijeme odkaz na hodnotu políčka, které je v tabulce dále od počátku (více vpravo nebo níže). Tento dopředný odkaz se projeví v okamžiku přepočtení tabulky.



Obrázek 4-4.
Demonstrace dopředných odkazů.

Uvažujme například políčko **e3** na obrázku 4-4. Tabulku máme rozdělenu na čtyři části - kvadranty, za střed bereme naše zvolené políčko.

Pokud se výraz v políčku **e3** odkazuje na políčka z kvadrantu I, bude výsledek vždy správný a to nezávisle na smyslu přepočítávání.

Pokud se výraz bude odkazovat na políčka z kvadrantu II, bude již výsledek záviset na smyslu přepočítávání. Pokud bude formulář přepočítán po řádcích, dostaneme správný výsledek, pokud po sloupcích, dostaneme výsledek chybný. V tomto případě se nejdříve vypočte hodnota našeho políčka (z neplatných hodnot) a teprve potom se přepočtou políčka z kvadrantu II.

Obdobná situace je s odkazy do kvadrantu III. Zde při přepočtení po sloupcích bude výsledek správný a po řádcích chybný.

A nakonec odkazy do kvadrantu IV způsobí nezávislost na smyslu přepočítávání, ovšem výsledky budou chybné vždy!

Poznámka: odkazy do kvadrantů II a III na políčka ve stejném sloupci nebo řádku jako pole e3 jsou správné a chybu nezpůsobí.

Po tomto rozboru je nám již jasné, na co si musíme při tvorbě formuláře dát pozor a u kterých formulářů musíme dodržovat jeden ze smyslů přepočítávání.

U některých formulářů s dopřednými odkazy dostaneme správné výsledky po opakovaném přepočtení, kdy po prvním přepočtení bude v diskutovaném políčku chybná hodnota ale do políčka, do kterého míří dopředný odkaz, se dostane správná hodnota. Při druhém přepočtení se tedy již bude počítat se správnou hodnotou. Ovšem to je zaručeno pouze u formulářů s jednou úrovní dopředných odkazů. U složitějších formulářů těchto odkazů může být více úrovní a přepočtení by bylo nutné provádět mnohokrát. Je tedy lepší snažit se vytvářet formuláře bez dopředných odkazu, i když to není vždy možné.

Ukažme si nyní příklady s dopřednými odkazy. V prvním příkladu si vytvoříme formulář se čtrnáctinásobnou úrovní dopředných odkazů a ve druhém příkladu použijeme dopředný

odkaz pro přenos výsledku z jednoho období jako základ dalšího období.

Vložme do políčka o1 číselnou hodnotu 1. Do políčka a1 vložíme vzorec

b1+1.

Nejedná se o cyklický odkaz, neboť se odkazujeme na jiné políčko než do kterého vzorec vkládáme. Tento vzorec zopakujeme až do políčka n1.

Po prvním přepočtení formuláře dostaneme výsledky jako na obrázku 4-5. Opakujme přepočtení. Hodnoty v prvním sloupci se budou zvyšovat po 1. Teprve po 14-tém přepočtení dostaneme v políčku a1 správnou hodnotu - číslo 15!

	1	2	3
a	2.00		
b	2.00		
c	2.00		
d	2.00		
e	2.00		
f	2.00		
g	2.00		
h	2.00		
i	2.00		
j	2.00		
k	2.00		
l	2.00		
m	2.00		
n	2.00		
o	1.00		

Obrázek 4-5.
Formulář se 14ti násobným dopředným odkazem po prvním přepočtení. Pouze v políčku n1 je správný výsledek.

Vytvořme nyní formulář podle obrázku 4-6. V políčku **o1** je číselná konstanta - základ (přenos z minulého roku). V políčku **a2** je vzorec pro překopírování tohoto základu

o1.

Do políček b2-e2 jsme pomocí opakování vložili stejný vzorec

INT (10000*RND),

který nám bude dávat náhodná čísla pro výpočet.

Nakonec jsme pomocí globální funkce vložili dvakrát sloupec na pozici druhého sloupce, takže jsme dostali tři sloupce se stejnými vzorci a různými náhodně vytvořenými čísly.

	1	2	3	4
a	PRENOS:	100000	126127	148306
b		5557	9096	3143
c		6798	2217	5790
d		9918	6325	4285
e		3854	4441	1390
f		-------	-------	-------
g	100000	126127	148206	162814

Obrázek 4-6.
Formulář, který používá dopředný odkaz pro přenos výsledků předchozího období jako základu dalšího období.

Nyní formulář přepočteme po sloupcích. Dostanete jiné výsledky než jsou uvedeny na obrázku 4-6 (použití generátoru náhodných čísel), ovšem výsledky budou v pořádku.

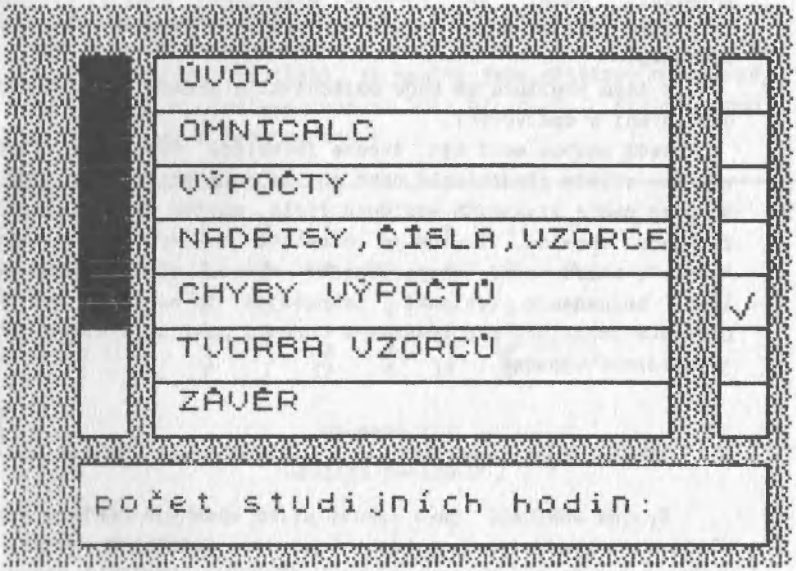
Přepočtěte opět formulář, tentokrát ale po řádcích. Porovnejte přenos základu mezi jednotlivými sloupci

g1 = a2
g4 = a3
g3 = a4.

Kromě prvního přenosu (přenos konstanty) budou všechny přenosy chybné! Nejdříve se totiž provedou přenosy a potom teprve se sečtou sloupce.

Vidíme, že i u formuláře se zdánlivě "rozumnou" strukturou mohou činit dopředné odkazy problémy. U formulářů tohoto typu je nutné dodržovat smysl přepočítávání. Je dokonce vhodné napsat do formuláře poznámku typu

PŘEPOČET PO SLOUPCÍCH!
POČÍTEJ PO ŘÁDCÍCH!


ÚVOD
OMNICALC
VÝPOČTY
NADPISY,ČÍSLA,VZORCE
CHYBY VÝPOČTŮ
TVORBA VZORCŮ
ZÁVĚR
počet studijních hodin:

5. Rozpoznávání, prevence a oprava chyb

Při práci s OMNICALCem, stejně jako s jinými počítačovými systémy, musíme si být jisti, že dosažené výsledky jsou správné. Musíme se sami sebe ptát, jestli námi vytvořený formulář pracuje správně a jestli tak bude pracovat i nadále. Musíme mít na mysli, že k chybám při práci s počítači dochází naší vinou a podnikat kroky k snížení pravděpodobnosti vzniku těchto chyb.

Za tímto účelem musíme dobře znát, jak k chybám při použití OMNICALCu může dojít. Z tohoto hlediska nás zajímají tyto části systému: mikropočítač, OMNICALC, námi připravený formulář, data, se kterými pracujeme, způsob použití formuláře (námi nebo jinými osobami) a finální použití výsledků.

V této kapitole se bude pojednávat o typech chyb, jejich odhalování a opravování.

Naší snahou musí být tvorba formulářů vymyšlených tak, aby se chybám předcházelo nebo aby se samy detekovaly. Pokud se máme podle získaných výsledků řídit, musíme si být jisti, že jsou správné. Z tohoto hlediska se vyplatí i práce "navíc", která není bezpodmínečně nutná, ale zajistí nám lepší bezpečnost výsledků. Samozřejmě to má smysl jen při přípravě formulářů pro opakované budoucí použití a nikoli pro jednorázový výpočet.

5.1. Vlastní prostředky OMNICALCu k předcházení chybám

Systém OMNICALC jako takový přímo obsahuje některé rysy sloužící k předcházení chybám. Jsou to například dotazy a zvuková návěští. Například kdykoliv, kdy OMNICALC očekává zadání funkce, objeví se na displeji text

funkce? (bdgjkmnoprstuvz).

Tento text nám napovídá, že můžeme zadat funkci a která písmena k tomu můžeme použít. Funkce se volí počátečním písmenem jejich názvu a navíc tyto písmena máme vypsány na displeji. Kdykoliv stiskneme klávesu, která v daném okamžiku nemá význam, ozve se hluboké zabručení, které nás upozorní, že něco děláme špatně. Pokud má zvolená funkce další parametry, OMNICALC se nás na ně průběžně ptá.

5.2. Základní chyby

Dobře si prohlédněte obrázek 5-1. Je na něm formulář sčítající v několika sloupcích tři číselné údaje, v řádku **f** jsou tyto součtu. Formulář obsahuje množství chyb. Již při letmém pohledu je vidět, že součty jsou většinou nesprávné. Ovšem i součty správné nemusí být bez chyby! Projděme nyní celý formulář a najděme chyby.

	1	2	3	4	5	6	7	8	9	10
a	1	2	3	4	55	6	7	8		
b										
c	6	6	0	4	6	4	5	9		
d	2	2	2	3	2	10	9			
e	5	5	5	2	7	7	12			
f	12	7	7	13	4	11	0			

Obrázek 5-1.
Množství příkladů chyb, se kterými se můžeme setkat při použití OMNICALCu.

Rozpoznávání, prevence a oprava chyb

Sloupec 1

Vzorec v políčku f1 je takovýto

(c1+c1+c3).

Jednoduše to je chybný vzorec pro součet tří čísel ve sloupci. Jednoduchost zápisu výrazů v OMNICALC může také způsobit problémy pro snadnost vložení chybného vzorce.

Sloupec 2

Opět chybný vzorec

d2+e2.

Je pravděpodobné, že chyba vznikla tak, že jsme nejprve vytvořili formulář s pouze dvěma řadami čísel. tehdy byl vzorec vložen dobře. Později jsme vložili nový řádek čísel ale již jsme neopravili součtový vzorec.

Sloupec 3

Konečně správný výsledek! Pro jistotu se podíváme na vzorec a zjistíme, že vzorec

d3+e3

je také chybný. Součet je stejný pouze proto, že v políčku c3 je 0. Tato chyba je o to záludnější, že se momentálně neprojevuje - je to latentní chyba. Teprve po změně hodnoty c3 se projeví.

Z tohoto příkladu je vidět, že je vhodné předem vyzkoušet připravený formulář s různými čísly. To je důležité zejména tehdy, když se použití chybných výsledků projeví za dlouhou dobu.

Sloupec 4

Políčko f4 je zadáno jako sloupcový součet. Ovšem tento součet sčítá všechna čísla nad sebou, tedy i číslo 4, kterým jsme si označili sloupec. V takto označeném formuláři nelze součtový typ políčka použít.

Sloupec 5

V políčku **f5** vidíme

4

Podivné. Vždyť do tohoto políčka jsme určitě ukládali správný vzorec pro součet

c5+d5+e5,

a najednou je zde místo vzorce číslo 4!?

Vzpomeneme si, že jsme zadávali čísla do sloupce: 6, 2, 7 a 4. Aha! Číslo 4 již mělo přijít do dalšího sloupce na pozici c6. Tam jsme ho opravdu znovu vložili, ale smazaný vzorec jsme již neopravili.

Je zřejmé, že je třeba věnovat zvýšenou pozornost při zápisu čísel - vyplňování formuláře. Jiné systémy mají k dispozici funkce umožňující "zamknout" všechna políčka se vzorci, do kterých se nesmí psát. Toto ale v OMNICALCu možné

není. Proto musíme sami podnikat kroky k zamezení těchto chyb, jako bylo například vytečkování zadávaných políček v prázdném formuláři.

Sloupec 6

Toto je divné, neboť vzorec v políčku f6 je správný

c6+d6+e6.

Chyba je v políčku d6. Zde číslo 10 zapsáno jako **text**, nikoli jako číslo. Text se potom vyhodnotí jako 0, nikoli 10.

Sloupec 7

Opět je zadán správný vzorec

c7+d7+e7.

Také čísla ve sloupci jsou zadána správně. Chyba je v tom, že jsme po změně čísel zapomněli formulář přepočítat. Po přepočtení již bude vše v pořádku.

Sloupec 8

Tato přebytečná 9 vznikla při zadávání, kdy jsme se z políčka c8 posunuli doprava a nikoli dolů. Chybě jsme předešli správným doplněním 9 do správného políčka d8, ovšem zapomněli jsme smazat políčko c9.

Uvedené příklady nám jasně ukázaly potřebu pečlivosti a opatrnosti při používání OMNICALCu. Vyvodíme z toho, že používání OMNICALCu je efektivní, ale musíme přitom dávat pozor na chyby, které za nás počítač hlídat nemůže.

Ve většině uvedených příkladů jsme vyřešili příklad chybně. V případech chybných vzorců to bylo jasné, ty také vždy kontrolujeme jako první. Další chyby se odhalovaly velice nesnadno. Proto než je hledat je lepší jim předcházet pečlivou prací.

Pamatujme si, že chyba mikropočítače nebo OMNICALCu je teoreticky možná, ale v naprosté většině případů si problémy s chybami způsobíme sami.

Zvláštní skupinou chyb jsou chyby způsobené naší zapomětlivostí. Například neopravení vzorců při doplnění další řádky dat. Pro součty je výhodnější používat součtová políčka, protože jejich funkce je nezávislá na počtu řádek. Ale pozor, abychom nesčítali i nechtěné údaje, jako tomu bylo v našem příkladu.

5.3. Ještě několik typických chyb

Ukažme si ještě další chyby, které nejsou specialitou OMNICALCu ale jsou vlastní všem počítačovým systémům.

Již víme, že když napíšeme 12345678901234 do políčka tabulky, je uloženo 12345678900000. Kdo pracuje s počítači o tom ví, je mu jasné že každý systém pracuje pouze s určitou přesností. Co je důležité, že v paměti je uloženo špatné (jiné) číslo.

Můžete si vyzkoušet, že

12345678901234-12345678900000

je správně 0, ale

123▮5678901234-123▮5678000000

je 901120 místo správného výsledku 900000.

Jinou ukázkou podivného výsledku je takovýto součet

```
      1
      1
      1
     ==
součet:  4
```

Toto je způsobeno tím, že se zobrazují pouze čísla, zatímco výpočet probíhá s plnou přesností třeba takto

1.1+1.8+1.4 = 4.3

5.4. Detekce a prevence chyb

Naším hlavním problémem je zisk přesných informací. Přestože chceme maximálně využít rychlosti práce s elektronickou tabulkou, nechceme to za cenu správnosti. Jak jsme viděli na příkladech, při práci nejvíce potřebuje pečlivost. Je velice snadné zadat chybný vzorec nebo ho přepsat číslem a velice těžké chybu zjistit a opravit, zejména u rozsáhlých formulářů.

Když tvoříme vlastní programy nebo používáme jiné systémy než OMNICALC, můžeme velkou část programu věnovat kontrole vstupních dat pro zajištění jejich správnosti. Pokud třeba máme spočítat výplaty v závislosti na odpracovaných hodinách, chceme mít jistotu, že zadané údaje odpovídají

skutečnosti a že se z nich vypočítají správné výsledky. Je výhodné, pokud zadávané údaje mají určitá omezení, například musí být větší nebo menší než nějaká hodnota. Například třeba nesmí pracovníci odpracovat více než 80 hodin měsíčně.

Musíme předem upozornit, že tyto kontroly neumí provádět sám OMNICALC ale že je musíme vymýšlet a zadávat sami.

Počítejme jednoduchou účetní uzávěrku, tak jak je vidět na obrázku 5-2.

	1	2	3	4	5
a		**Účetní uzávěrka**			
b					
c		1	5	0	10
d		1	5	0	10
e		1	5	0	10
f		1	5	0	10
g		-----	-----	-----	-----
h	**celkem:**	4	20	0	40
i					
j		1	5	0	10
k		1	5	0	10
l		-----	-----	-----	-----
m	**celkem:**	2	10	0	20
n					
o		=====	=====	=====	=====
p	**CELKEM:**	6	30	0	60

Obrázek 5-2.
Typická ukázka formuláře s testovacími daty ale bez dvojité kontroly.

Nyní si ukážeme jednoduchou možnost dvojité kontroly. Pracovní součty jsou počítány pomocí podsoučtových políček a celkový součet je počítán součtovým polem. Ovšem tento celkový součet můžeme také počítat jako součet podsoučtů

h2+▮2

a obdobně v dalších sloupcích. Přitom výsledek samozřejmě musí být stejný. Jinou možností by byl vzorec pro součet jednotlivých políček

c2+d2+e2+f2+j2+k2

Nebo je možné sečíst celý sloupec včetně podsoučtů a podělit výsledek dvěma

(a2+b2+c2+d2+e2+f2+g2+h2+i2+j2+k2+▮2▪▪2)/2

Jednu z těchto možností použijeme a vložíme ji do řádky **r**. Do řádky **s** vložíme vzorec pro porovnání výsledků

p2-r2

Výsledek je na obrázku 5-3. Nyní stačí zkontrolovat, jestli v řádku s jsou všude nuly, případně si ještě vytvořit políčko pro součet této řádky. Pokud se někde objeví jiné číslo než 0, máme ve formuláři chybu.

Při tisku můžeme tisk pomocných dvou řádek vynechat.

	1	2	3	4	5
a		Účetní uzávěrka			
b					
c		1	5	0	10
d		1	5	0	10
e		1	5	0	10
f		1	5	0	10
g		-----	-----	-----	-----
h	celkem:	4	20	0	40
i					
j		1	5	0	10
k		1	5	0	10
l		-----	-----	-----	-----
m	celkem:	2	10	0	20
n					
o		=====	=====	=====	=====
p	CELKEM:	6	30	0	60
q					
r		6	30	0	60
s	kontrola:	0	0	0	0

Obrázek 5-3.
Formulář z obrázku 5-2 po doplnění kontrolními vzorci.

Ukažme si nyní jinou techniku, která umožňuje testovat vstupní data. Vezměme si třeba problém výpočtu mzdy. Je dána hodinová mzda a počet odpracovaných hodin jednotlivých pracovníků. Počet hodin odpracovaných za jeden měsíc je shora omezen 80 hodinami. Potřebujeme tedy nějakou funkci, která by nám umožnila indikaci překročení stanoveného limitu. V jiných systémech se k tomuto účelu používá vestavěné funkce MAX (maximum). Výraz

```
MAX(a1-80,0)
```

má hodnotu nula, pokud je a1 (počet hodin) menší než 80. Při

překročení tohoto limitu má výraz hodnotu počtu hodin o které je limit překročen.

V programu OMNICALC funkce MAX není k dispozici, takže si musíme pomoci jinak. Použijeme vlastností logické funkce AND a zkusíme tento vzorec

```
a1-80 AND a1   80
```

Dokud je **a1** menší než 80, porovnání na pravé straně neplatí a **x AND 0** je vždy nula. Při překročení limitu porovnání začne platit a výsledkem **x AND 1** je **x**, tedy počet hodin nad limit.

Na obrázku 5-4 je příklad použití této kontroly.

	1	2	3
a	jméno	hodin	test
b	Novák	5	0
c	Smejkal	70	0
d	Smutný	90	10

Obrázek 5-4.
Příklad použití kontrolního sloupečku pro test správnosti vstupních dat.

5.5. Oprava chyb

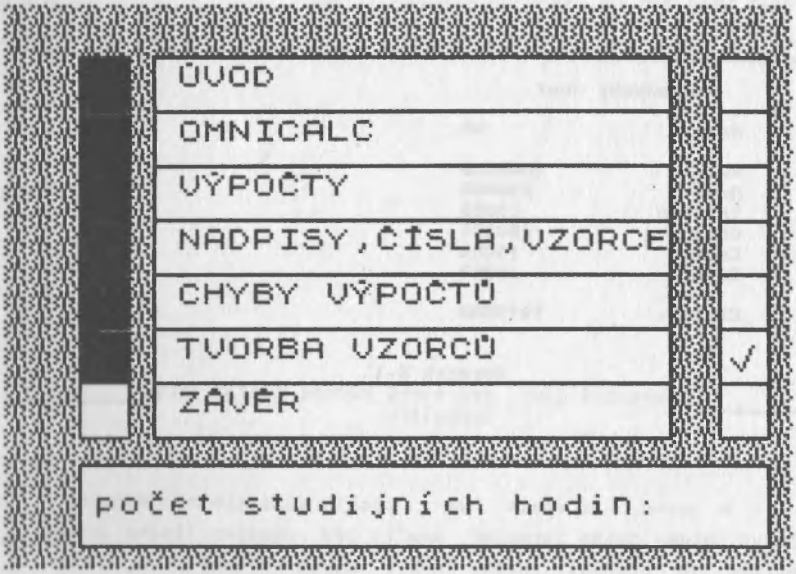
Pokud se nám při testování projeví chyba, musíme najít a opravit její příčinu. Testování musí být natolik pečlivé, abychom si byli jisti, že ve formuláři nejsou žádné chyby. To nám usnadní hledání chyb při výpočtu - budou to pouze chyby způsobené nepozorností při zadávání číselných hodnot. Také je třeba dávat pozor na to, abychom při opravě chyb nešťastně

nezpůsobili chyby jiné. Po opravení nalezených chyb opět přistoupíme k testování.

Ladění, proces lokalizace a oprav chyb ve formuláři, obyčejně začíná v políčku, kde je viditelně chybný výsledek. Pokud se toto políčko ukáže jako bezchybné, pokračujeme kontrolou políček, na které se odkazujeme ve výrazu počátečního políčka. Tímto postupem krok za krokem šťastně vyřešíme náš problém. Pokud nalezneme chybu vzniklou při globálních změnách nebo opakování vzorce, budou nejspíše chyby ve všech políčcích se shodným vzorcem.

5.6. Shrnutí

OMNICALC nám při popisovaných operacích ladění nemůže poskytnout žádnou "inteligentní" podporu. Jsme odkázáni na ruční práci krok za krokem. Nemůžeme získat seznam referencí - nemůžeme jednoduše zjistit všechna políčka, která se odkazují na nějaké políčko. Nemůžeme zjistit, která políčka tabulky jsou prázdná a která obsahují údaje jinak než pohybem okénka po celé tabulce. Nemůžeme si nechat zobrazit v políčcích výrazy, které obsahují a nikoli pouze číselný výsledek. S těmito užitečnými funkcemi se můžeme setkat u jiných systémů ale u OMNICALCu ne.


ÚVOD
OMNICALC
VÝPOČTY
NADPISY,ČÍSLA,VZORCE
CHYBY VÝPOČTŮ
TVORBA VZORCŮ
ZÁVĚR
počet studijních hodin:

6. Tvorba vzorů

Hlavní schopností a účelem OMNICALCu je vytváření **vzorů**. Vzor je elektronický formulář, do kterého jsme vepsali texty, hlavičky, vzorce a některé potřebné číselné údaje. Tento vzor si uložíme a později, po získání potřebných číselných hodnot, si ho načteme, doplníme čísly a okamžitě dostaneme výsledky.

Podívejme se na jednoduchý účet na obrázku 6-1. Při tomto pohledu nemůžeme říci nic o vztazích mezi údaji. Jak jsou vypočteny mzdy? Jsou funkcí počtu osob, které pro nás pracují? Závisí daně na mzdách, počtu osob nebo na obojím? Kolik počítáme na cestovné pro jednu osobu?

Jednoduchý účet

Osob	**50**
Mzdy	**1000000**
Daně	**250000**
Telefon	**10000**
Odvod	**200000**
Cesty	**15000**
Různé	**18000**
CELKEM:	**1493000**

Obrázek 6-1.
Jednoduchý účet, pro který budeme tvořit vzor formuláře.

Je jasné, že výpis nám o těchto vztazích nic neřekne. Pokud chceme tento formulář použít pro výpočet jiného účtu, můžeme to provést pouze tehdy, pokud přesně víme, která políčka a jak změnit. Musíme znát, která čísla a vzorce změnit a které ponechat. Pokud jsme formulář připravili před

několika měsíci, pravděpodobně si již žádné detaily nebudeme pamatovat.

Vytvoření prázdného vzoru nám pomůže odstranit tyto problémy. Vytvořme vzor pro účet z obrázku 6-1. Příprava tohoto vzoru bude probíhat v několika odlišných krocích.

6.1. Analýza problému

Jednoduchost práce s OMNICALCem nás nabádá k okamžitému započetí vyplňování prázdné tabulky vzorci a čísly. To je ale chyba. U nejjednodušších příkladů to sice možné je, ale u složitějších by tento postup byl zdrojem problémů.

Rozeberme si nejdříve pravidla pro výpočet jednotlivých položek.

políčko	typ	popis
OSOB	proměnná	Toto políčko je proměnná, to znamená, že do něj vpisujeme vstupní číslený údaj. Při vytváření vzoru do tohoto políčka vepíšeme značku pro vložení čísla.
MZDY	výraz	Tuto hodnotu počítáme jako součin počtu osob a průměrného ročního platu 20000. Do vzoru vepíšeme tento vzorec.
DANĚ	výraz	Daně počítáme jako 25% z mezd.
TELEFON	proměnná	Platba za telefon je číslo, které musíme zadat před výpočtem.
ODVOD	konstanta	I když OMNICALC nerozlišuje mezi proměnnou a konstantou, pro tvorbu vzoru je nutné tyto typy rozlišovat. Konstantu vložíme přímo do vzoru, neboť je vždy stejná.
CESTY	výraz	Počítáme jako 300 na osobu.
RŮZNÉ	proměnná	Zadává se před výpočtem.
CELKEM	výraz	Součet všech uvedených položek.

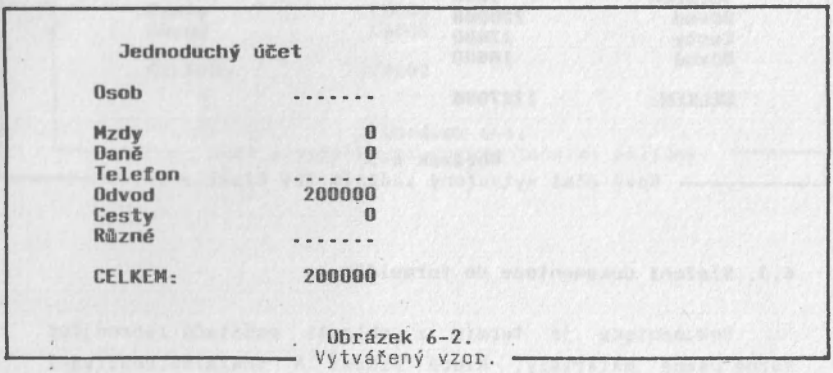
Nyní podle tohoto rozboru vytvoříme vzor formuláře. Na místa, kam se mají zapsat číselné hodnoty, vložíme tečky (.....). Toto je pouze návrh značení, můžete si vytvořit svou vlastní symboliku, která vám bude více vyhovovat.

6.2. Symbolika pro identifikaci typu políček

Při vytváření vzoru vkládáme vzorce, nadpisy a konstanty přímo do tabulky. V políčcích, do kterých jsme vložili vzorce, se objeví nuly. Jak víme, OMNICALC vyhodnocuje textová políčka jako nulu. Například do políčka MZDY byl vložen vzorec pro součin počtu osob * 20000. V políčku OSOB je text ..... a ten je vyhodnocen jako 0. Součin

20000 * 0

dává nulu, tak jak je vidět na obrázku 6-2. V tomto případě je vyhodnocování textů jako 0 velmi výhodné.



Obrázek 6-2.
Vytvářený vzor.

Tento vzor je již přehlednější než účet z obrázku 6-1. Políčka, do kterých máme zadat údaje jsou zřetelně označeny tečkami.

Na obrázku 6-3 je účet, který jsme vytvořili zadáním tří čísel do vzoru. Vytvořili jsme ho následujícím postupem

načtení vzoru

- nastavení kurzoru na tři políčka s tečkami pro vložení hodnot
- zadání těchto hodnot:

OSOB : 40
TELEFON : 5000
RŮZNÉ : 10000

Jednoduchý účet

Osob	**40**
Mzdy	**800000**
Daně	**200000**
Telefon	**5000**
Odvod	**200000**
Cesty	**12000**
Různé	**10000**
CELKEM:	**1227000**

Obrázek 6-3.
Nový účet vytvořený zadáním tří čísel.

6.3. Vložení dokumentace do formuláře

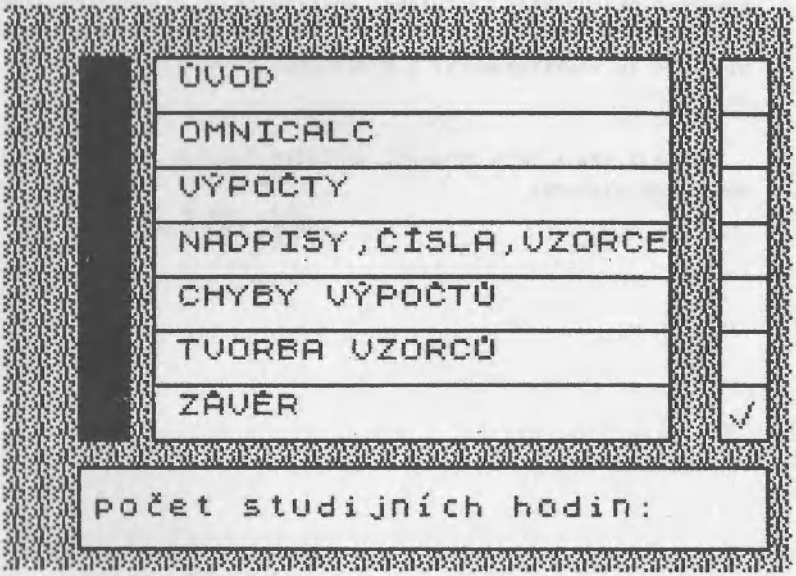
Dokumentace je termín z oblasti počítačů zahrnující různé psané materiály, které slouží k snazšímu používání počítačového systému nebo jeho části. Může se jednat o dokumentaci k programu, popis práce, systémová dokumentace, uživatelská dokumentace atd. V našem případě se bude jednat o dokumentaci sloužící uživateli vzoru formuláře.

Množství pomocných informací záleží na způsobu práce. Pokud použijeme vzor několikrát v krátkém časovém období, postačí nám pár základních informací. Ovšem pokud se vzor bude používat na různých místech dlouhou dobu, je nutná detailní dokumentace.

Na obrázku 6-4 je ukázka takovéto dokumentace.

```
Vzor:                ZÁKL-OC-1/87
Autor:               Ing. Vít Libovický
Datum vytvoření:     30.9.1987
Datum použití:       1.11.1987
Uživatel:            Ing. Petr Novák

                  Jednoduchý účet

Osob                      40

Mzdy                  800000
Daně                  200000
Telefon                 5000
Odvod                 200000
Cesty                  12000
Různé                  10000

CELKEM:               1227000
```

Obrázek 6-4.
Účet s vyplněnými dokumentačními políčky.


ÚVOD
OMNICALC
VÝPOČTY
NADPISY,ČÍSLA,VZORCE
CHYBY VÝPOČTŮ
TVORBA VZORCŮ
ZÁVĚR
✓
počet studijních hodin:

7. Závěr

Na stránkách této učebnice jsme se pokusili naznačit aplikace z některých oblastí lidské činnosti, ve kterých může OMNICALC pomoci při řešení problémů a dále pak metody, které jsou k této činnosti zapotřebí. Pro omezený rozsah této knihy nebylo možné ukázat množství hotových formulářů z různých oblastí aplikace OMNICALCu, ale věříme, že i z těch několika popsaných jste pochopili základní pravidla, kterými je nutné se při práci řídit.

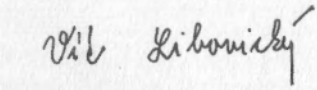
Jako ve všech oblastech práce s počítači, pro důkladné pochopení této problematiky je nejlepší praxe. Zkuste si sami vymýšlet nejrůznější problémy, analyzujte je a vytvořte vzor formuláře pro jejich řešení. Postupujte od jednoduchých problémů ke složitějším až k praktickým úlohám z oblasti vaší práce.

Přeji vám v této činnosti množství dobrých nápadů a samé bezchybné výsledky

Vít Libovický

Autor

Materiál připravil Mikrokomputer klub 666. ZO SVAZARMu
Poštovní schránka 64, 169 00 Praha 69



Autor: Ing. Vít Libovický

Sazba : Ing. Petr Novák
Grafická úprava obalu: ak. mal. Karel Čapek

Texty pořízeny na mikropočítači **Video Genie EG 3003** textovým procesorem **SUPERSCRIPSIT**. Předlohy pro tisk zpracovány textovým procesorem **WordPerfect 4.2** na systému **IBM PC/XT** a **ROBOTRON S 6011**. Na tvorbu obrázků byl použit grafický editor **ART-STUDIO** na mikropočítači **ZX SPECTRUM** ve spolupráci s mozaikovou tiskárnou **STAR GX-10**. Technické informace pro zpracování materiálů touto metodou poskytne mládežnická komplexní racionalizační brigáda **"Sova 3"** Centra pro mládež vědu a techniku ÚV SSM.

Kopírování povoleno pouze pro osobní nebo studijní účely.