

Jiří Barilla, Pavel Šimr

Microsoft® **Excel** pro techniky a inženýry

**Řešení
každodenních úkolů
v technické praxi**

Využití funkcí
ve výpočtech
a tvorba grafů

Statistické zpracování
a evidence dat

Optimalizace
výroby v Excelu

 **C PRESS**



Jiří Barilla, Pavel Simr

Microsoft Excel **pro techniky a inženýry**

Computer Press, a.s.
Brno
2008

Microsoft Excel pro techniky a inženýry

Jiří Barilla, Pavel Simr

Computer Press, a.s., 2008. Vydání první.

Jazyková korektura: Petra Láníčková

Vnitřní úprava: Martina Petrová

Sazba: Martina Petrová

Rejstřík: Pavel Simr

Obálka: Martin Sodomka

Komentář na zadní straně obálky: Libor Pácl

Technická spolupráce: Jiří Matoušek, Petr Klíma,
Dagmar Hajdajová, Zuzana Šindlerová

Odpovědný redaktor: Libor Pácl

Technický redaktor: Jiří Matoušek

Produkce: Daniela Nečasová

Computer Press, a.s.,

Holandská 8, 639 00 Brno

Objednávky knih:

<http://knihy.cpress.cz>

distribuce@cpress.cz

tel.: 800 555 513

ISBN 978-80-251-2421-5

Prodejní kód: K1172

Vydalo nakladatelství Computer Press, a.s., jako svou 3114. publikaci.

© Computer Press, a.s. Všechna práva vyhrazena. Žádná část této publikace nesmí být kopírována a rozmnožována za účelem rozšiřování v jakékoli formě či jakýmkoli způsobem bez písemného souhlasu vydavatele.

Obsah

Úvod	11
Komu je kniha určena	11
Uspořádání knihy	11
Typografická konvence použita v knize	12
1	
Vybraná témata z Excelu pro techniky	13
Vzorce a funkce pro techniky	14
Vytvoření jednoduchého vzorce	14
Vytvoření technického vzorce s využitím funkcí	15
Relativní, absolutní a smíšená adresace buněk ve vzorcích a funkcích	15
Vložení funkce v Excelu	16
Vybrané funkce a jejich použití	18
Funkce SUMA	18
Funkce PRŮMĚR	18
Funkce MIN	18
Funkce POČET	19
Funkce POČET2	19
Funkce COUNTIF	19
Funkce COUNTIFS	19
Funkce ABS	20
Funkce ODMOCNINA	20
Funkce PI	20
Funkce SIN	20
Funkce COS	21
Funkce ZAOKROUHLIT	21
Funkce ROK	21
Funkce SUMIF	21
Funkce SUBTOTAL	22
Funkce KDYŽ	22
Funkce SVYHLEDAT	23
Funkce VVYHLEDAT	23
Funkce INDEX	24
Funkce POSUN	25
Funkce VAR	26
Funkce SMODCH	26
Funkce PRŮMODCHYLKA	26
Funkce COVAR	26
Funkce CORREL	27
Funkce LINREGRESE	27
Funkce SOUČIN.SKALÁRNÍ	27
Maticové vzorce	28
Ověřování vstupních dat	29
Podmíněné formátování	29

Nástroje pro analýzu dat	30
Hledání řešení	30
Řešitel	30
Scénáře	33
Kovariance	35
Korelace	35
Regrese	35
Jednotný vzhled sešitu a práce se seznamy	35
Motivy	36
Použití motivu	36
Barvy motivů	36
Znaková sada motivů	37
Efekty motivů	38
Uložení a odstranění vlastního motivu	38
Styly tabulky	38
Práce se seznamy	39
Základy maker a VBA	40
Vytvoření makra	40
Vytvoření vlastní funkce ve VBA	42
Deklarace funkce	42
Název funkce	43
Parametry funkcí	43
Aktivace Editoru Visual Basicu	43

2

Grafická znázornění dat 45

Typy grafů	46
Zásady tvorby grafů	48
Terminologie	49

Tvorba technických grafů	50
Jednoduchý graf	50
Graf výrobního plánu	50
Přidání dat do jednoduchého grafu	51
Kombinování různých typů grafů	52
Časová řada	52
Přerušená datová řada	53
Graf s vedlejší (druhou) osou Y	54
Histogram	55
Srovnávací histogram	57
Grafická analýza naměřených dat	58
Zrychlení na nakloněné rovině	59
Měření rezonanční křivky sériového rezonančního obvodu	60
Klouzavý průměr	62
Chybové úsečky	62
Zobrazení průběhu matematické funkce	63
Zobrazení funkce dvou proměnných	66
Směs ve výšečovém grafu	67

Máme hodně hodnot v grafu	69
Dynamické zobrazení nejnovějších dat z tabulky	70
Vytvoření odkazů na buňku	72
Vložení odkazu do názvu grafu	72
Formátování grafů	74
Změna kompozice grafu	75
Úprava velikosti měřítka	76
Stupnice grafu	77
Styly	77
Automatické vytváření grafů	78
Víceúrovňový popis kategorií	79
Mřížka grafu	80
Datové řady	81
Zmrazení grafu	81
Viditelnost grafů na listu	81

3

Technické výpočty v Excelu

83

Výpočty povrchů, objemů a hmotností technických těles	84
Vzorový příklad na výpočet povrchu, objemu a hmotnosti obdélníkového bazénu	84
Interpretace výsledků	87
Vytvoření vlastní funkce pro výpočet objemu bazénu ve VBA	87
Vytvoření vlastní funkce pro výpočet povrchu bazénu ve VBA	88
Výpočet objemu a povrchu bazénu pomocí vlastních funkcí	89
Výpočet hloubky bazénu pomocí nástroje Hledání řešení	90
Interpretace výsledků	90
Analýza výsledků s pomocí scénáře	91
Interpretace výsledků	93
Příklad výpočtu povrchu, objemu a hmotnosti zkoseného obdélníkového bazénu	93
Interpretace výsledků	96
Vytvoření vlastní funkce pro výpočet objemu bazénu ve VBA	96
Vytvoření vlastní funkce pro výpočet povrchu bazénu ve VBA	97
Výpočet objemu a povrchu bazénu pomocí vlastních funkcí	97
Výpočet délky bazénu pomocí nástroje Hledání řešení	98
Interpretace výsledků	99
Analýza výsledků pomocí scénáře	99
Interpretace výsledků	100
Příklad výpočtu povrchu, objemu a hmotnosti kruhového bazénu	100
Interpretace výsledků	102
Vytvoření vlastní funkce pro výpočet objemu bazénu ve VBA	103
Vytvoření vlastní funkce pro výpočet povrchu bazénu ve VBA	103
Výpočet objemu a povrchu bazénu pomocí vlastních funkcí	104
Výpočet průměru bazénu pomocí nástroje Hledání řešení	105
Interpretace výsledků	105
Analýza výsledků pomocí scénáře	105
Interpretace výsledků	106
Příklad výpočtu povrchu, objemu a hmotnosti železné koule	107

Interpretace výsledků	108
Vytvoření vlastní funkce pro výpočet objemu koule ve VBA	108
Vytvoření vlastní funkce pro výpočet povrchu koule ve VBA	109
Výpočet objemu a povrchu koule pomocí vlastních funkcí	110
Výpočet průměru koule pomocí nástroje Hledání řešení	111
Interpretace výsledků	111
Analýza výsledků pomocí scénáře	111
Interpretace výsledků	112

Technologické výpočty **112**

Vzorový příklad technologických výpočtů pro provoz a údržbu zkoseného obdélníkového bazénu	113
Interpretace výsledků	118
Vytvoření vlastní funkce pro výpočet množství tepla k ohřátí vody ve VBA	118
Vytvoření vlastní funkce pro výpočet množství přísady na úpravu pH vody ve VBA	119
Výpočet objemu, povrchu, množství tepla k ohřátí vody a množství přísady pro úpravu pH pomocí vlastních funkcí	120
Výpočet teploty vody pomocí nástroje Hledání řešení	121
Interpretace výsledků	122
Analýza výsledků pomocí scénáře	122
Interpretace výsledků	123
Hlídní doporučených hodnot pomocí podmíněného formátování	124

Konstrukční výpočty **125**

Vzorový příklad konstrukčních výpočtů pro zhotovení činky na posilování	125
Výpočet povrchu a hmotnosti kotouče	127
Výpočet povrchu a hmotnosti hřídele	128
Výpočet povrchu a hmotnosti činky	128
Interpretace výsledků	130
Vytvoření vlastní funkce pro výpočet hmotnosti kotouče ve VBA	130
Vytvoření vlastní funkce pro výpočet hmotnosti hřídele ve VBA	131
Výpočet hmotnosti kotouče a hřídele pomocí vlastních funkcí	131
Výpočet průměru kotouče pomocí nástroje Hledání řešení	132
Interpretace výsledků	133
Analýza výsledků pomocí scénáře	133
Interpretace výsledků	134

4

Evidence technických dat **135**

Evidence majetku **137**

Vzorový příklad vytvoření evidence majetku	137
Vyhledávání údajů v databázovém seznamu	139
Seřazení údajů v databázovém seznamu	140
Vytváření souhrnů	141
Interpretace výsledků	143
Vytváření skupin	145
Vytváření součtů ve skupinách	146
Vytvoření kontingenční tabulky	148
Vytvoření kontingenčního grafu	150

Výběr záznamů pomocí automatického filtru	152
Vytvoření součtu vybraných záznamů pomocí funkce SUBTOTAL	154
Vytvoření součtu pomocí funkce SUMIF	156
Zaokrouhlování číselných hodnot	157
Označení číselných hodnot pomocí podmíněného formátování	159
Zjištění počtu záznamů pomocí funkce COUNTIF	160
Vyhledávání údajů pomocí funkce SVYHLEDAT	162
Vyhledávání údajů pomocí funkce VVYHLEDAT	167
Aktualizace kontingenční tabulky a grafu pomocí maker	171

Evidence zakázek **172**

Vzorový příklad vytvoření evidence zakázek	173
Vyhledávání údajů v databázovém seznamu	174
Seřazení údajů v databázovém seznamu	174
Vytváření souhrnů	175
Interpretace a analýza výsledků	176
Interpretace a analýza výsledků	180
Vytvoření kontingenční tabulky	180
Vytvoření kontingenčního grafu	183
Využití automatického filtru a funkce SUBTOTAL	184
Vytvoření součtu pomocí funkce SUMIF	186
Zjištění počtu záznamů pomocí funkce COUNTIFS	186
Sledování plnění zakázek pomocí funkce KDYŽ	188

Evidence technické dokumentace **190**

Vzorový příklad vytvoření evidence technické dokumentace	190
Vyhledávání údajů v databázovém seznamu	190
Seřazení údajů v databázovém seznamu	191
Výběr záznamů pomocí automatického filtru	192
Vytvoření kontingenční tabulky	192
Zjištění počtu záznamů pomocí funkce COUNTIFS	194
Sledování zápůjček pomocí funkce KDYŽ	195

5

Statistické zpracování dat **199**

Statistické zpracování naměřených dat **200**

Vzorový příklad statistického zpracování naměřených dat	201
Výpočet průměrné (střední) hodnoty	202
Výpočet průměrné (střední) hodnoty absolutních odchylek	204
Výpočet rozptylu	204
Výpočet směrodatné odchylky	205
Interpretace a analýza výsledků	206
Grafické znázornění rozptylu naměřených hodnot kolem průměrné hodnoty	206
Příklad porovnání dvou dávkovačů cementu	207
Výpočet průměrné (střední) hodnoty	208
Výpočet průměrné (střední) hodnoty absolutních odchylek	209
Výpočet rozptylu	209
Výpočet směrodatné odchylky	209
Interpretace a analýza výsledků	210

Grafické znázornění rozptylu naměřených hodnot kolem průměrné hodnoty	210
Výpočet základních statistických charakteristik pomocí běžných funkcí a vzorců	212
Výpočet průměrné hodnoty	212
Výpočet rozdílu, absolutní hodnoty a druhé mocniny	213
Výpočet průměrné (střední) hodnoty absolutních odchylek	214
Výpočet rozptylu	214
Výpočet směrodatné odchylky	214
Výpočet rozdílu, absolutní hodnoty a druhé mocniny pomocí maticových vzorců	214
Zjišťování závislostí mezi několika naměřenými soubory	215
Vzorový příklad zjišťování závislostí naměřených dat	217
Výpočet kovariance	218
Výpočet korelace	218
Interpretace a analýza výsledků	219
Grafické znázornění závislosti naměřených dat	219
Zjišťování závislosti naměřených dat pomocí běžných funkcí a vzorců	220
Výpočet průměrné hodnoty a směrodatné odchylky	220
Výpočet kovariance	222
Výpočet korelace	222
Výpočet rozdílu a součinu pomocí maticových vzorců	223
Příklad vyšetření závislosti spotřeby topného oleje	223
Výpočet kovariance	224
Výpočet korelace	225
Interpretace a analýza výsledků	225
Grafické znázornění závislosti spotřeby topného oleje na teplotě	225
Grafické znázornění závislosti spotřeby topného oleje na velikosti obytné plochy	226
Řešení pomocí nástroje Analýza dat	226
Proložení experimentálních dat odpovídající funkcí	228
Vzorový příklad proložení experimentálních dat odpovídající funkcí	229
Výpočet kovariance a korelace	230
Interpretace výsledků	231
Grafické znázornění závislosti Y na X	231
Proložení naměřených dat odpovídající funkcí	231
Interpretace a analýza výsledků	234
Grafické znázornění rozptylu naměřených hodnot	235
Výpočet hodnoty funkce a druhé mocniny rozdílu pomocí maticových vzorců	236
Řešení pomocí funkce LINREGRESE	236
Interpretace výsledků	238
Řešení pomocí nástroje Analýza dat	239
Interpretace výsledků	240
Příklad na vyšetření závislosti spotřeby paliva u automobilu na jeho rychlosti	240
Výpočet kovariance a korelace	241
Interpretace výsledků	242
Grafické znázornění závislosti Y na X	242
Proložení naměřených dat odpovídající funkcí	243
Interpretace a analýza výsledků	245
Grafické znázornění rozptylu naměřených hodnot	246
Výpočet hodnoty funkce a druhé mocniny rozdílu pomocí maticových vzorců	247

6

Matematické modelování v technické praxi

249

Modely výrobních procesů	250
Formulace modelu výrobního procesu	251
Vzorový příklad vytvoření matematického modelu výroby	252
Ekonomický model	254
Matematický model a jeho řešení	256
Ekonomická interpretace a analýza výsledků	262
Analýza výsledků za pomoci scénáře	264
Řešení pomocí funkce KDYŽ	266
Řešení pomocí maticových vzorců	267
Modely přepravy zboží	268
Vzorový příklad vytvoření matematického modelu přepravy zboží	269
Ekonomický model	269
Matematický model a jeho řešení	270
Ekonomická interpretace a analýza výsledků	279
Řešení pomocí funkce KDYŽ	279
Řešení pomocí skalárního součinu	281
Optimalizační modely	282
Formulace optimalizačního modelu	282
Vzorový příklad optimalizace výroby	284
Ekonomický model	284
Matematický model a jeho řešení	285
Ekonomická interpretace a analýza výsledků	289
Analýza výsledků pomocí scénáře a uložení nastavení Řešitele	291
Řešení pomocí maticových vzorců	294
Kapacitní úloha s vnitřní vazbou	295
Ekonomický model	295
Matematický model a jeho řešení	296
Ekonomická interpretace a analýza výsledků	301
Řešení pomocí maticových vzorců	305

7

Optimalizace v Excelu

307

Kapacitní úlohy	308
Úlohy o optimálním dělení materiálů	308
Směšovací úlohy	308
Distribuční úlohy	308
Optimalizace výroby	309
Úloha na dosažení minimálních nákladů na výrobu	309
Ekonomický model	310
Matematický model a jeho řešení	311
Ekonomická interpretace a analýza výsledků	320
Řešení pomocí maticových vzorců	321

Optimální rozřezání tyčí a prken na požadované rozměry	322
Úloha na optimální rozřezání tyčí	323
Ekonomický model	323
Matematický model a jeho řešení	324
Ekonomická interpretace a analýza výsledků	330
Řešení pomocí maticových vzorců	332
Optimální namíchání různých směsí	332
Úloha na sestavení krmné směsi	333
Ekonomický model	333
Matematický model a jeho řešení	334
Ekonomická interpretace a analýza výsledků	339
Řešení pomocí maticových vzorců	340
Optimalizace přemísťování objektů	340
Úloha na přemístění strojů	341
Ekonomický model	341
Matematický model a jeho řešení	342
Ekonomická interpretace a analýza výsledků	348
Řešení pomocí skalárního součinu	348
Optimalizace přepravy zboží	349
Úloha na přepravu písku	349
Ekonomický model	350
Matematický model a jeho řešení	351
Ekonomická interpretace a analýza výsledků	357
Řešení pomocí skalárního součinu	358

Rejstřík	361
-----------------	------------

Úvod

Excel je jeden z nejrozšířenějších programů, který je využíván ve firmách i pro soukromou potřebu. V Excelu jsou zpracovávány různé databázové seznamy, statistické a technické výpočty apod. Znalost Excelu je vyžadována téměř při všech výběrových řízeních pracovníků na ekonomické a technické profese. Tato kniha je napsaná pro Microsoft Excel 2007. U nižších verzí Excelu budou odlišnosti pouze ve způsobu používání nabídek.

Komu je kniha určena

Kniha je určena zejména technikům, kteří chtějí Excel využívat nejen pro tvorbu databázových seznamů, ale také pomocí Excelu provádět technické výpočty a statistické zpracování dat. Kniha obsahuje kapitoly, které mohou posloužit i čtenářům, kteří nejsou technicky zaměřeni (například manažerům). Použití vybraných funkcí a nástrojů Excelu je vysvětleno na praktických příkladech.

Uspořádání knihy

Kniha je členěna do jednotlivých kapitol, které tvoří samostatný celek. Pouze kapitola 7 navazuje na kapitolu 6. Kapitoly 1 a 2 lze chápat jako úvodní kapitoly, ve kterých jsou vybraná témata z Excelu pro techniky. Některá témata v těchto kapitolách byla zpracována na základě knih o Excelu, jejichž autorem je Milan Brož, kterému patří poděkování za inspiraci k napsání prvních dvou kapitol.

V *první kapitole* naleznete popis všech funkcí a analytických nástrojů, které jsou využívány při řešení praktických úloh. Jsou to zejména:

- vzorce a funkce,
- analytické nástroje,
- práce se seznamy,
- šablony motivů tabulek,
- základy maker a VBA.

Druhá kapitola se zabývá grafickým zobrazením dat. Je zaměřena zejména na tvorbu technických grafů, volbu měřítek a formátování. Tvorba grafů je vysvětlena na praktických příkladech.

Ve třetí kapitole jsou na příkladech z technické praxe ukázány možnosti provádění technických výpočtů v Excelu. Při technických výpočtech jsou využívány vzorce, funkce a analytické nástroje. Pro pokročilejší uživatele je ukázáno využití vlastních funkcí ve VBA.

Čtvrtá kapitola se věnuje evidenci technických dat. V každé firmě se evidují data, která jsou pro ni důležitá. Malé a střední firmy používají často k evidenci dat Excel, který poskytuje velké množství prostředků pro práci s databázovými seznamy. V této kapitole je na praktických příkladech ukázáno využití důležitých databázových funkcí a nástrojů.

V páté kapitole je ukázána možnost statistického zpracování naměřených dat. Jsou v ní vysvětleny:

- základní statistické charakteristiky pro zpracování technických dat,
- základní statistické charakteristiky pro zjišťování závislostí mezi několika naměřenými soubory,
- základní statistické charakteristiky pro proložení experimentálních dat odpovídající funkcí.

Tyto statistické charakteristiky jsou využity při řešení praktických příkladů.

Šestá kapitola se zabývá využitím matematických modelů v technické praxi. Pomocí matematických modelů lze získávat nezbytné informace o pracovním procesu a na základě těchto informací se správně rozhodovat. Tvorba matematických modelů je ukázána na příkladech z praxe.

Sedmá kapitola navazuje na kapitolu šestou a je zaměřena na optimalizační modely. Pomocí optimalizačních modelů lze:

- najít optimální výrobní program,
- optimálně nařezat materiál pro výrobu (například tyče),
- optimálně namíchat různé směsi,
- optimálně přepravit stroje na nová stanoviště,
- optimálně přepravit zboží od dodavatelů k odběratelům.

V této kapitole jsou řešeny praktické příklady ze všech výše uvedených oblastí.

Předpokládané znalosti. Kniha se nezabývá výukou základů Excelu – je určená technikům, kteří mají základní znalost Excelu. Z oblasti matematiky jsou předpokládány znalosti ze střední školy.

Poděkování patří spoluautorovi Pavlu Simrovi, který vypracoval druhou kapitolu a vyhotovil schémata k praktickým příkladům.

I přes péči, která byla věnována tvorbě této publikace, nelze vyloučit možnost výskytu chyb. Autor proto nepřebírá žádné záruky ani právní odpovědnost za využití uvedených informací a z toho plynoucích důsledků.

Veškeré osoby a jména uvedená v této knize jsou pouze ilustrativní a fiktivní a jakákoliv podobnost s osobami žijícími je čistě náhodná. V knize jsou použity zjednodušené praktické příklady, které mají výukový charakter. V příkladech jsou použita modelová data.

Jiří Barilla

Typografická konvence použitá v knize

V celé příručce je použito toto grafické členění:

Tučné písmo Prvky grafického uživatelského rozhraní (příkazy, tlačítka apod.)

Kurzíva Důležité výrazy v rámci textu.

Speciální symboly:



Poznámky



Tipy



Důležitá upozornění a varování

1

Vybraná témata z Excelu pro techniky

V této kapitole:

Vzorce a funkce pro techniky

Nástroje pro analýzu dat

Jednotný vzhled sešitu a práce se seznamy

Základy maker a VBA

Pro řešení praktických úloh z technické praxe je nezbytné mít potřebné znalosti z určitých oblastí Excelu. Pro techniky jsou důležité zejména tyto oblasti:

- vzorce a funkce,
- analytické nástroje,
- práce se seznamy,
- šablony motivů tabulek,
- základy maker a VBA.

K těmto základním oblastem pak můžeme přidat celou řadu dalších.

Hlavní důraz bude kladen na vytváření technických vzorců a práci s vybranými funkcemi, které budeme využívat při řešení praktických úloh v jednotlivých kapitolách knihy.

Vzorce a funkce pro techniky

Vzorce jsou základním nástrojem Excelu pro techniky, protože umožňují provádět technické výpočty. Excel má velké množství různých funkcí, které můžeme využívat samostatně nebo v kombinaci se vzorci.

Rozdíl mezi vzorcem a funkcí je v tom, že vzorec si vytváříme sami, kdežto funkci máme k dispozici jako hotový produkt v Excelu. S funkcí pracujeme tak, že zadáme její název a argumenty funkce.

Praktické příklady využití vzorců a funkcí najdeme ve všech kapitolách této knihy při řešení praktických úloh.

Vytvoření jednoduchého vzorce

Vzorec zapíšeme do zvolené buňky tak, že:

1. Označíme buňku, do které chceme zapsat vzorec.
2. Zapíšeme znak = (rovná se).
3. Zapíšeme číslo nebo adresu buňky.
4. Zapíšeme jeden z následujících matematických operátorů:
 - + pro sčítání (součet),
 - - pro odečítání (rozdíl),
 - * pro násobení (součin),
 - / pro dělení (podíl),
 - ^ pro mocninu.
5. Zapíšeme další číslo nebo adresu buňky.



Poznámka: Bod 4 a 5 pak opakujeme podle potřeby.

Vytvoření vzorce si ukážeme na jednoduchém příkladě, ve kterém máme sečíst dvojnásobek buňky A1 s trojnásobkem buňky B1. Vzorec chceme umístit do buňky C1.

Budeme postupovat tak, že:

1. Označíme buňku C1, do které chceme zapsat vzorec.
2. Zapišeme znak = (rovná se).
3. Zapišeme číslo 2.
4. Zapišeme matematický operátor *.
5. Buď zapišeme, nebo vytyčením zadáme adresu buňky A1.
6. Zapišeme matematický operátor +.
7. Zapišeme číslo 3.
8. Zapišeme matematický operátor *.
9. Buď zapišeme, nebo vytyčením zadáme adresu buňky B1.

Po potvrzení zápisu vzorce klávesou **Enter** se do buňky C1 zapiše vzorec:

$=2*A1+3*B1$. (1.1)

Vytvoření technického vzorce s využitím funkcí

Ukážeme si zápis vzorce, který má umocnit součet buněk A1 a A2 na druhou a k tomu přičíst druhou odmocninu z podílu buněk A3 a A4. Vzorec do zvolené buňky zapišeme ve tvaru:

$=(A1+A2)^2+ODMOCNINA(A3/A4)$. (1.2)



Poznámky:

- Funkci můžeme do vzorce buď zapsat, nebo zadat pomocí dialogu **Argumenty funkce**.
- Příklady využití technických vzorců jsou ve všech kapitolách této knihy.

Relativní, absolutní a smíšená adresace buněk ve vzorcích a funkcích

Jednou z největších výhod vzorců a funkcí je to, že se při kopírování mění (účelově) relativní adresy buněk. To má tu výhodu, že pokud chceme například sečíst čísla ve sloupcích tabulky, pak nám stačí vložit vzorec (popř. funkci) pouze jednou a potom jej zkopírovat do vedlejších buněk.

Buňky lze adresovat v rámci jednoho sešitu dvěma způsoby:

- V aktivním listu (list, se kterým pracujeme) tvoří adresu buňky název (souřadnice) sloupce a číslo (souřadnice) řádku, například A1.
- V jiném listu je před názvem sloupce název listu, například List1!A1.

Relativní adresa buňky je tvořena pouze názvem sloupce a číslem řádku (například A1) a má tyto vlastnosti:

- Při kopírování vzorce v řádku se mění název sloupce (například =A1, =B1, =C1 atd.).
- Při kopírování vzorce ve sloupci se mění číslo řádku (například =A1, =A2, =A3 atd.).

Absolutní adresa buňky je tvořena tak, že před názvem sloupce a číslem řádku je znak \$ (například =\$A\$1). Při kopírování se adresa buňky nemění.

Smišovaná adresa buňky je tvořena tak, že je absolutně adresován buď sloupec, nebo řádek (ne oba současně), například $=\$A1$ nebo $=A\$1$. Při kopírování vzorce se mění pouze relativní člen (souřadnice) adresy.



Tip: Absolutní adresu ve vzorci můžeme zadat také tak, že:

1. Vytyčením zadáme adresu buňky.
2. Stiskneme klávesu **F4**.

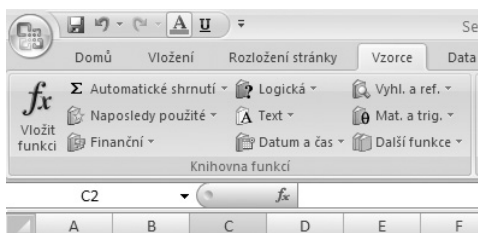
Smišovou adresu ve vzorci zadáme opakovaným stiskem klávesy **F4**.

Vložení funkce v Excelu

Funkce je hotový vzorec v Excelu, který můžeme využívat tak, že zadáme její název a argumenty funkce. Například zápis $=\text{SUMA}(A1:A5)$ znamená, že použijeme funkci **SUMA**, která sečte všechna čísla v oblasti buněk A1:A5.

Funkci můžeme zadat čtyřmi základními způsoby:

- zápisem funkce do buňky,
- na kartě **Vzorce** ve skupině **Knihovna funkcí** pomocí nabídky **Vložit funkci**,
- na kartě **Vzorce** ve skupině **Knihovna funkcí** pomocí nabídky **Automatické shrnutí**,
- klepnutím na tlačítko f_x , které je umístěno vedle řádku vzorců (viz obrázek 1.1).



Obrázek 1.1 Možnosti pro vložení funkce

Do buňky zapišeme funkci tak, že:

1. Označíme buňku, do které chceme zapsat funkci.
2. Zapišeme název funkce a její argumenty, například:

$=\text{SUMA}(A1:A5)$.

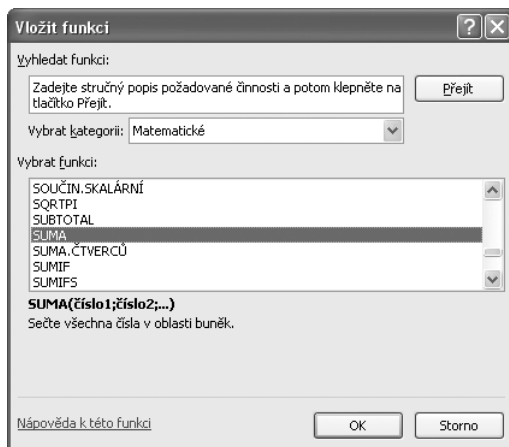
(1.3)



Důležité: Při zápisu funkce musíme začít znakem $=$ (rovná se) stejně jako při zápisu vzorce.

Pomocí nabídky **Vložit funkci** zadáme funkci tak, že:

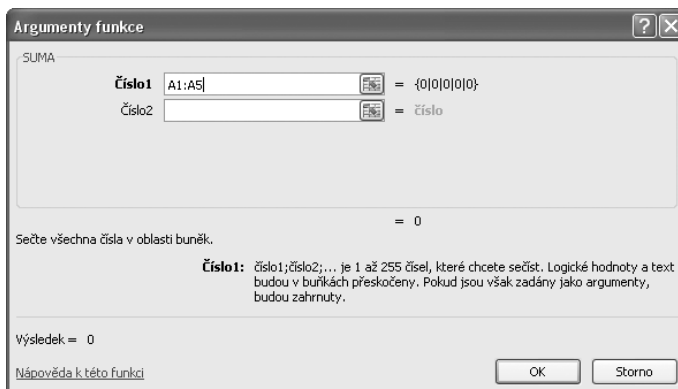
1. Označíme buňku, do které chceme zapsat funkci.
2. Zvolíme kartu **Vzorce**.
3. Ve skupině **Knihovna funkcí** klepneme na položku **Vložit funkci**.
4. V dialogu **Vložit funkci** vybereme v rozbalovací nabídce **Vybrat kategorii:** Matematické.



Obrázek 1.2 Dialog Vložit funkci

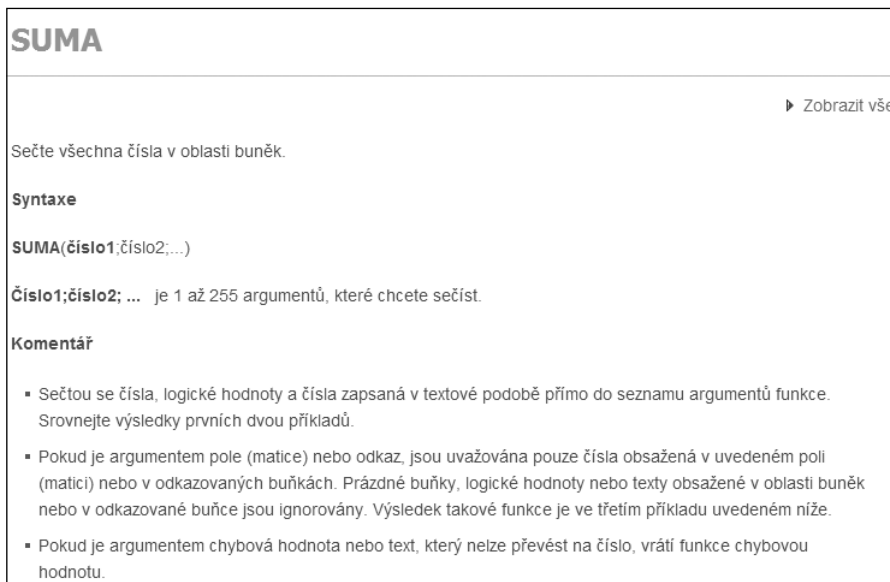
5. V okně **Vybrat funkci** označíme funkci **SUMA** (viz obrázek 1.2).
6. V dialogu **Argumenty funkce** v okně **Číslo1** vtyčením zadáme oblast buněk A1:A5 (viz obrázek 1.3).
7. Potvrdíme **OK**.

Informace o použití této funkce najdeme v nápovědě, kterou si zobrazíme tak, že:



Obrázek 1.3 Dialog Argumenty funkce

1. Zvolíme kartu **Vzorce**.
2. Ve skupině **Knihovna funkcí** klepneme na položku **Vložit funkci**.
3. V dialogu **Vložit funkci** vybereme v rozbalovací nabídce **Vybrat kategorii**: Matematické.
4. V okně **Vybrat funkci** označíme funkci **SUMA**.
5. V levém dolním rohu klepneme na text: **Nápověda k této funkci** (viz obrázek 1.4).



Obrázek 1.4 Nápověda k funkci SUMA

Pomocí nabídky **Automatické shrnutí** zadáme funkci tak, že:

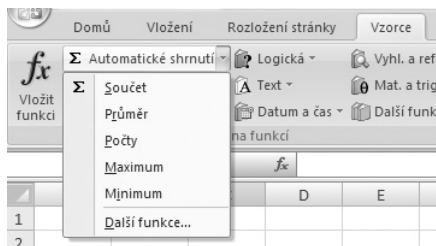
1. Označíme buňku, do které chceme vložit funkci.
2. Zvolíme kartu **Vzorce**.

3. Ve skupině **Knihovna funkcí** klepneme na rozbalovací tlačítko položky **Automatické shrnutí**.
4. Vybereme jednu z možností na obrázku 1.5.



Poznámka:

- Pokud vybereme jeden z názvů funkce (Součet, Průměr, Počty, Maximum nebo Minimum) na obrázku 1.5, vloží se do buňky funkce bez zadáných argumentů, například pro Součet se vloží funkce `=SUMA()`, do které zadáme vytčením oblast, kterou chceme sečíst.
- Pokud vybereme nabídku **Další funkce**, zobrazí se dialog **Vložit funkci** (viz obrázek 1.2), s jehož pomocí si vybereme potřebnou funkci.
- Pokud klepneme na tlačítko **Automatické shrnutí**, vloží se do buňky funkce **SUMA**.



Obrázek 1.5 Nabídka Automatické shrnutí

Klepnutím na tlačítko f_x se zobrazí dialog **Vložit funkci** (viz obrázek 1.2), s jehož pomocí si vybereme potřebnou funkci.

Vybrané funkce a jejich použití

Není možné se v této knize zabývat všemi funkcemi, protože Excel jich obsahuje velké množství. Vybereme pouze ty funkce, které budeme využívat v jednotlivých kapitolách při řešení praktických úloh. V této kapitole uvedeme pouze jejich stručnou charakteristiku s tím, že s praktickým použitím funkce se můžeme seznámit v kapitole, ve které je daná funkce použita při řešení příkladu.

Funkce SUMA

Funkce **SUMA** je jednou z nejčastěji používaných funkcí v Excelu. Slouží k sečtení všech čísel z vybrané oblasti buněk. Zápis funkce:

`=SUMA(číslo1;číslo2;...)`.

Číslo1;číslo2; ... je 1 až 255 argumentů (oblastí), které chceme sečíst.

Funkce **SUMA** je použita v kapitolách 3, 4, 5, 6 a 7.

Funkce PRŮMĚR

Funkce **PRŮMĚR** slouží k vypočítání aritmetického průměru z vybrané oblasti buněk. Zápis funkce:

`=PRŮMĚR(číslo1;číslo2;...)`.

Číslo1;číslo2; ... je 1 až 255 argumentů (oblastí), ze kterých chceme vypočítat průměr.

Funkce **PRŮMĚR** je použita v kapitole 2 a 5.

Funkce MIN

Funkce **MIN** slouží k nalezení minimální hodnoty z vybrané oblasti buněk. Zápis funkce:

`=MIN(číslo1;číslo2;...)`.

Číslo1;číslo2; ... je 1 až 255 argumentů (oblastí), ze kterých chceme nalézt minimální hodnotu.

Funkce **MIN** je použita v kapitole 6.

Funkce POČET

Funkce **POČET** slouží ke zjištění počtu čísel ve vybrané oblasti buněk. Zápis funkce:

`=POČET(hodnota1;[hodnota2];...)`.

- **hodnota1** je povinný argument. První položka, odkaz na buňku nebo oblast, ve které chceme spočítat čísla.
- **hodnota2; ...** je nepovinný argument. Až 255 dalších položek, odkazů na buňky nebo oblasti, ve kterých chceme spočítat čísla.

Funkce **POČET** je použita v kapitole 5.

Funkce POČET2

Funkce **POČET2** slouží ke zjištění počtu buněk ve vybrané oblasti, které nejsou prázdné. Zápis funkce:

`=POČET2(hodnota1;[hodnota2];...)`.

- **hodnota1** je povinný argument. První položka, odkaz na buňku nebo oblast, v níž chceme spočítat buňky, které nejsou prázdné.
- **hodnota2; ...** je nepovinný argument. Až 255 dalších položek, odkazů na buňky nebo oblasti, v nichž chceme spočítat buňky, které nejsou prázdné.

Funkce **POČET2** je použita v kapitole 2.

Funkce COUNTIF

Funkce **COUNTIF** spočítá počet buněk v oblasti, které splňují jedno zadané kritérium. Můžeme například spočítat všechny buňky začínající určitým písmenem nebo všechny buňky obsahující číslo větší či menší než zadané číslo. Zápis funkce:

`=COUNTIF(oblast;kritérium)`.

- **oblast** je povinný argument. Jedna nebo více buněk pro provedení výpočtu, včetně čísel či názvů, polí nebo odkazů obsahujících čísla. Prázdné hodnoty a textové hodnoty jsou ignorovány.
- **kritérium** je povinný argument. Číslo, výraz, odkaz na buňku nebo textový řetězec, které definují buňky, jež mají být započítány. Kritérium může mít například následující podobu: 32, „>32“, B4, „jablka“ nebo „32“.

Funkce **COUNTIF** je použita v kapitole 4.

Funkce COUNTIFS

Funkce **COUNTIFS** spočítá počet buněk v oblasti, které splňují více zadaných kritérií. Můžeme například spočítat všechny buňky začínající určitým písmenem nebo všechny buňky obsahující číslo větší či menší než zadané číslo. Zápis funkce:

=COUNTIFS(oblast_kritérií1;kritérium1;[oblast_kritérií2;kritérium2]...).

- **oblast_kritérií1** je povinný argument. První oblast, ve které mají být vyhodnocena přidružená kritéria.
- **kritérium1** je povinný argument. Kritérium vyjádřené číslem, výrazem, odkazem na buňku nebo textem, které definuje buňky, jež mají být započítány. Kritérium může mít například následující podobu: 32, „>32“, B4, „jablka“ nebo „32“.
- **oblast_kritérií2, kritérium2, ...** je nepovinný argument. Další oblasti a jejich přidružená kritéria. Je možné zadat až 127 dvojic oblast/kritérium.



Důležité: Každá další oblast musí mít stejný počet řádků a sloupců jako argument **oblast_kritérií1**. Oblasti spolu nemusí sousedit.

Funkce **COUNTIFS** je použita v kapitole 4.

Funkce ABS

Funkce **ABS** vypočítá absolutní hodnotu čísla. Absolutní hodnota čísla je totéž číslo bez znaménka (například absolutní hodnota čísla -2 je 2). Zápis funkce:

=ABS(číslo).

Číslo je reálné číslo, jehož absolutní hodnotu chceme zjistit.

Funkce **ABS** je použita v kapitole 3 a 5.

Funkce ODMOCNINA

Funkce **ODMOCNINA** vypočítá druhou odmocninu daného čísla. Zápis funkce:

=ODMOCNINA(číslo).

Číslo je číslo, jehož odmocninu chceme vypočítat.

Funkce **ODMOCNINA** je použita v kapitole 3 a 5.

Funkce PI

Funkce **PI** vloží do buňky nebo vzorce číslo 3,14159265358979, matematickou konstantu pí (Ludolfovo číslo), s přesností na 15 platných číslic. Zápis funkce:

=PI().

Funkce **PI** je použita v kapitole 2 a 3.

Funkce SIN

Funkce **SIN** vypočítá sinus daného úhlu. Zápis funkce:

=SIN(číslo).

Číslo je úhel v radiánech, jehož sinus chceme vypočítat.

Funkce **SIN** je použita v kapitole 2.

Funkce COS

Funkce **COS** vypočítá kosinus daného úhlu. Zápis funkce:

`=COS(číslo).`

Číslo je úhel v radiánech, jehož kosinus chceme vypočítat.

Funkce **COS** je použita v kapitole 2.

Funkce ZAOKROUHLIT

Funkce **ZAOKROUHLIT** zaokrouhlí číslo na zadaný počet číslic. Zápis funkce:

`=ZAOKROUHLIT(číslo;čísllice).`

- **číslo** je povinný argument. Jde o číslo, které chceme zaokrouhlit.
- **čísllice** je povinný argument. Určuje počet číslic, na které chceme číslo zaokrouhlit.

Funkce **ZAOKROUHLIT** je použita v kapitole 4.

Funkce ROK

Funkce **ROK** převede zadané datum na rok (například datum 21.5.2005 převede na 2005). Rok je celé číslo v rozmezí 1900-9999. Zápis funkce:

`=ROK(pořadové).`

Pořadové je datum, které chceme převést na rok.

Funkce **ROK** je použita v kapitole 4.

Funkce SUMIF

Funkce **SUMIF** vypočítá součet hodnot v oblasti buněk, které splňují určité kritérium buď ve stejné, nebo v jiné oblasti buněk. Zápis funkce:

`=SUMIF(oblast;kritéria;součet).`

- **oblast** je povinný argument. Jde o oblast buněk vyhodnocovanou pomocí daného kritéria. Buňky v jednotlivých oblastech musí představovat čísla nebo názvy, matice či odkazy, které obsahují čísla. Prázdné a textové hodnoty jsou ignorovány.
- **kritéria** je povinný argument. Jde o kritérium vyjádřené číslem, výrazem, odkazem na buňku, textem nebo funkcí, která definuje buňky, jež mají být sečteny. Kritérium může mít například následující podoby: 32, „>32“, B5, „32“, „jablka“ nebo DNES().



Důležité: Textová kritéria nebo kritéria obsahující logické či matematické symboly musí být uzavřena v uvozovkách („“). U číselných kritérií nejsou uvozovky nutné.

- **součet** je nepovinný argument. Jde o buňky, které budou sečteny v případě, že chceme sečíst jiné buňky, než jaké jsou zadány v argumentu *oblast*. Pokud je argument *součet* vynechán, sečte Excel buňky zadané v argumentu *oblast* (tedy buňky, u kterých je použito zadané kritérium).

Funkce **SUMIF** je použita v kapitole 4.

Funkce SUBTOTAL

Funkce **SUBTOTAL** vypočítá souhrn dat v seznamu nebo v databázi. Je vhodná zejména pro souhrny u vybraných dat pomocí automatického filtru. Zápis funkce:

=SUBTOTAL(konstanta_funkce;odkaz1;odkaz2; ...).

- **Konstanta_funkce** je číslo od 1 do 11 (zahrnuje skryté hodnoty) nebo od 101 do 111 (ignoruje skryté hodnoty), které určuje typ funkce použité při výpočtu souhrnů v seznamu (viz tabulka 1.1).
- **Odkaz1, odkaz2** atd. je 1 až 254 oblastí nebo odkazů, u kterých má být proveden souhrn.

Funkce **SUBTOTAL** je použita v kapitole 4.

Tabulka 1.1 Význam konstanty funkce SUBTOTAL

Konstanta_funkce (zahrnuje skryté hodnoty)	Konstanta_funkce (ignoruje skryté hodnoty)	Funkce
1	101	PRŮMĚR
2	102	POČET
3	103	POČET2
4	104	MAX
5	105	MIN
6	106	SOUČIN
7	107	SMODCH.VÝBĚR
8	108	SMODCH
9	109	SUMA
10	110	VAR.VÝBĚR
11	111	VAR

Funkce KDYŽ

Funkce **KDYŽ** vrátí určitou hodnotu, pokud je zadaná podmínka vyhodnocena jako **PRAVDA**, a jinou hodnotu, pokud je zadaná podmínka vyhodnocena jako **NEPRAVDA**. Funkce **KDYŽ** se používá při testování hodnot a vzorců. Zápis funkce:

=KDYŽ(podmínka;ano;ne).

- **Podmínka** je libovolná podmínka nebo výraz, který může být vyhodnocen jako PRAVDA nebo NEPRAVDA. Například A10 = 100 je logický výraz. Pokud má buňka A10 hodnotu 100, je tento výraz vyhodnocen jako PRAVDA. V opačném případě je vyhodnocen jako NEPRAVDA.
- **Ano** je hodnota, která je vrácena, jestliže hodnota argumentu podmínka je PRAVDA.
- **Ne** je hodnota, která je vrácena, jestliže hodnota argumentu podmínka je NEPRAVDA.



Poznámka: Při vytváření složitějších testů může být do sebe vnořeno jako argumenty ano a ne až 64 funkcí **KDYŽ**.

Funkce **KDYŽ** je použita v kapitole 4 a 6.

Funkce SVYHLEDAT

Funkce **SVYHLEDAT** vyhledá v prvním sloupci tabulky zadanou hodnotu a vrátí hodnotu odpovídající buňky ve stejném řádku jiného sloupce tabulky. Písmeno S v názvu funkce **SVYHLEDAT** znamená sloupec. Zápis funkce:

`=SVYHLEDAT(hledat, tabulka, sloupec, typ).`

Hledat: Hodnota, kterou chceme vyhledat v prvním sloupci. Může to být hodnota nebo odkaz. Pokud je hodnota hledat menší než nejmenší hodnota v prvním sloupci tabulky, vrátí funkce **SVYHLEDAT** chybovou hodnotu **#N/A**.

Tabulka je dva nebo více sloupců údajů. Je možné použít odkaz na oblast nebo název oblasti. Hodnoty v prvním sloupci tabulky jsou hodnoty prohledávané pomocí argumentu hledat. Tyto hodnoty mohou být textové, číselné nebo logické. Velká a malá písmena se nerozlišují.

Sloupec je číslo sloupce v oblasti určené parametrem tabulka, z něhož chceme vrátit odpovídající hodnotu. Pokud sloupec = 1, bude funkce vracet hodnotu z prvního sloupce tabulky. Pokud sloupec = 2, bude vracet hodnotu z druhého sloupce tabulky atd. Zadáte-li hodnotu argumentu sloupec:

- menší než 1, vrátí funkce **SVYHLEDAT** chybovou hodnotu **#HODNOTA!**,
- větší než počet sloupců v tabulce, vrátí funkce **SVYHLEDAT** chybovou hodnotu **#REF!**.

Typ je logická hodnota, která určuje, zda má funkce **SVYHLEDAT** nalézt přesnou nebo přibližnou hodnotu:

- Pokud má hodnotu PRAVDA nebo není zadán, bude vrácena přesná nebo přibližná shoda. Není-li přesná shoda nalezena, bude vrácena nejvyšší hodnota, která je menší než hodnota hledat. Hodnoty v prvním sloupci tabulky musí být seřazeny vzestupně, jinak nemusí funkce **SVYHLEDAT** vrátit správnou hodnotu.
- Pokud má hodnotu NEPRAVDA, vrátí funkce **SVYHLEDAT** pouze přesnou shodu. V tom případě nemusí být hodnoty v prvním sloupci tabulky seřazené. Obsahuje-li první sloupec tabulky dvě nebo více hodnot, které odpovídají argumentu hledat, bude použita první nalezená hodnota. Jestliže nebude nalezena přesná shoda, vrátí funkce chybovou hodnotu **#N/A**.

Funkce **SVYHLEDAT** je použita v kapitole 4.

Funkce VVYHLEDAT

Funkce **VVYHLEDAT** vyhledá danou hodnotu v horním řádku tabulky a vrátí hodnotu buňky z určeného řádku stejného sloupce. Funkce **VVYHLEDAT** se používá k vyhledávání hodnot v tabulce s nadepsanými sloupci. První písmeno V v názvu funkce **VVYHLEDAT** vyjadřuje, že funkce vyhledává hodnoty vodorovně (v řádcích). Zápis funkce:

`=VVYHLEDAT(hledat; tabulka; řádek; typ).`

Hledat je hodnota, kterou chceme vyhledat v prvním řádku tabulky. Může to být hodnota, odkaz nebo textový řetězec.

Tabulka je prohledávaná tabulka. Je možné použít odkazy na oblast nebo názvy oblastí.

- Hodnoty v prvním řádku tabulky mohou být textové, číselné nebo logické.

- Má-li argument typ hodnotu PRAVDA, musí být hodnoty prvního řádku tabulky vzestupně uspořádány: ...-2, -1, 0, 1, 2,... , A-Z, NEPRAVDA, PRAVDA; jinak funkce **VVYHLEDAT** může vrátit nesprávnou hodnotu.
- Pokud má argument typ hodnotu NEPRAVDA, první řádek tabulky nemusí být uspořádan.
- Funkce nerozlišuje malá a velká písmena.

Řádek je číslo řádku, ze kterého je vrácena odpovídající hodnota. Řádek s hodnotou 1 vrátí hodnotu z prvního řádku tabulky a řádek s hodnotou 2 vrátí hodnotu z druhého řádku tabulky. Má-li argument řádek hodnotu menší než 1, vrátí funkce **VVYHLEDAT** chybovou hodnotu #HODNOTA!. Má-li argument řádek hodnotu větší, než je počet řádků v oblasti tabulka, vrátí funkce **VVYHLEDAT** chybovou hodnotu #REF!.

Typ je logická hodnota, která určuje, zda funkce **VYHLEDAT** bude vyhledávat přesnou nebo přibližnou shodu.

- Má-li argument hodnotu PRAVDA nebo je-li vynechán, je vrácena hodnota přibližné shody. To znamená, že pokud nebyla nalezena přesná shoda, vrátí funkce největší možnou hodnotu, která je menší než hodnota argumentu hledat. Hodnoty v prvním řádku tabulky musí být seřazeny vzestupně, jinak nemusí funkce **VVYHLEDAT** vrátit správnou hodnotu.
- Je-li hodnota argumentu NEPRAVDA, bude funkce **VVYHLEDAT** hledat pouze přesnou shodu. Pokud shoda neexistuje, vrátí funkce chybovou hodnotu #N/A.

Funkce **VVYHLEDAT** je použita v kapitole 4.

Funkce INDEX

Funkce **INDEX** vrátí hodnotu nebo odkaz na hodnotu z tabulky nebo oblasti. Jsou dvě různé formy funkce **INDEX**: matice a odkaz.

Forma maticová: Funkce vrátí hodnotu (odkaz) nebo pole odkazů (hodnot) z jedné oblasti. Zápis funkce:

=INDEX(pole; řádek; sloupec).

Pole je oblast buněk nebo maticová konstanta.

- Pokud argument pole určuje oblast pouze s jedním řádkem nebo sloupcem, může být příslušný argument řádek nebo sloupec vynechán.
- Pokud oblast pole obsahuje více než jeden řádek a sloupec a je použit pouze jeden z argumentů řádek nebo sloupec, vrátí funkce **INDEX** celý řádek nebo sloupec.

Řádek určuje řádek pole. Pokud je argument řádek vynechán, je argument sloupec povinný.

Sloupec určuje sloupec pole. Pokud je argument sloupec vynechán, je argument řádek povinný.



Poznámky:

- Použijeme-li oba argumenty řádek a sloupec, vrátí funkce **INDEX** hodnotu buňky ležící v průsečíku zadaného argumenty řádek a sloupec.
- Zadáme-li do argumentů řádek nebo sloupec hodnotu 0 (nula), vrátí funkce **INDEX** pole (matici) hodnot celého sloupce nebo řádku.

- Argumenty řádek a sloupec musí odkazovat na buňku v rámci určené oblasti; jinak funkce **INDEX** vrátí chybovou hodnotu **#REF!**.

Forma odkazu: Funkce vrátí hodnotu (odkaz) nebo pole odkazů (hodnot) výběrem z více oblastí. Zápis funkce:

`=INDEX(odkaz;řádek;sloupec;oblast).`

Odkaz je odkaz na jednu či více oblastí buněk.

- Pokud se argument odkaz vztahuje na nesouvislou oblast, musí být zadán se závorkami.
- Pokud každá odkazovaná oblast obsahuje pouze jeden řádek nebo sloupec, je argument řádek nebo sloupec nepovinný. Odkazujeme-li například na jeden řádek, zadáme hodnotu `INDEX(odkaz;;sloupec)`.

Řádek určuje řádek, který se má protínat.

Sloupec určuje sloupec, který se má protínat.

Oblast určuje oblast, ve které má ležet průsečík. Oblasti jsou číslovány od 1. Pokud je argument oblast vynechán, použije funkce **INDEX** oblast číslo 1.



Poznámky:

- Po zpracování argumentu odkaz a oblast je určena konkrétní oblast, argumenty řádek a sloupec určují konkrétní buňku: řádek 1 je prvním řádkem určené oblasti, sloupec 1 je prvním sloupcem určené oblasti. Funkce **INDEX** vrátí průsečík určený argumentem řádek a sloupec.
- Má-li argument řádek nebo sloupec hodnotu 0, vrátí funkce **INDEX** odkaz na celý sloupec nebo řádek.
- Argumenty řádek, sloupec a oblast musí odkazovat na buňku v rámci určené oblasti, jinak funkce **INDEX** vrátí chybovou hodnotu **#REF!**. Pokud jsou argumenty řádek a sloupec vynechány, vrátí funkce **INDEX** odkaz na oblast určenou argumentem oblast.
- Výsledkem funkce **INDEX** je odkaz, který může být předložen jiným funkcím. V závislosti na vzorci vrátí funkce **INDEX** hodnotu, která může být použita jako hodnota nebo jako odkaz.

Funkce **INDEX** je použita v kapitole 2.

Funkce POSUN

Funkce **POSUN** vrátí odkaz na oblast, která obsahuje určený počet řádků a sloupců, od určité buňky nebo oblasti buněk. Vracený odkaz může být jedna buňka nebo oblast buněk. Počet řádků a sloupců, které se mají vrátit, můžeme určit. Zápis funkce:

`=POSUN(odkaz;řádky;sloupce;výška;šířka).`

- **Odkaz** je původní odkaz, vůči kterému provádíme posun. Pokud odkaz neodkazuje na buňku nebo oblast sousedících buněk, vrátí funkce **POSUN** chybovou hodnotu **#HODNOTA!**.
- **Řádky** je počet řádků, o které se má posunout levá horní buňka nového odkazu (nahoru nebo dolů). Zadáme-li například číslo 5, levá horní buňka odkazu bude pět řádků pod levou horní buňkou původního odkazu. Můžeme použít kladnou (dolů od původního odkazu) nebo zápornou hodnotu (nahoru od původního odkazu).

- **Sloupce** je počet sloupců vlevo nebo vpravo, o které se má posunout levá horní buňka výsledného odkazu vzhledem k původnímu odkazu. Zadáme-li například číslo 5, bude levá horní buňka odkazu o pět sloupců vpravo od levé horní buňky původního odkazu. Můžeme použít kladnou (posun doprava od původního odkazu) i zápornou hodnotu (posun doleva od původního odkazu).
- **Výška** je požadovaná výška (počet řádků) výsledného odkazu. Výška je vždy kladné číslo.
- **Šířka** je požadovaná šířka (počet sloupců) výsledného odkazu. Šířka je vždy kladné číslo.

Funkce **POSUN** je použita v kapitole 2.

Funkce VAR

Funkce **VAR** vypočítá rozptyl základního souboru. Zápis funkce:

`=VAR(číslo1;číslo2;...)`.

Číslo1, číslo2, ... je 1 až 255 argumentů, vztahujících se ke vzorku základního souboru.

Funkce **VAR** je použita v kapitole 5.

Funkce SMODCH

Funkce **SMODCH** vrátí směrodatnou odchylku základního souboru určenou z náhodného výběru. Směrodatná odchylka vyjadřuje, jak se hodnoty liší od průměrné hodnoty (střední hodnoty). Zápis funkce:

`=SMODCH(číslo1;číslo2;...)`.

Číslo1, číslo2, ... je 1 až 255 argumentů, vztahujících se ke vzorku základního souboru.

Funkce **SMODCH** je použita v kapitole 5.

Funkce PRŮMODCHYLKA

Funkce **PRŮMODCHYLKA** vrátí průměr absolutních odchylek bodů dat od jejich střední hodnoty. **PRŮMODCHYLKA** je měřítkem variability množiny dat. Zápis funkce:

`=PRŮMODCHYLKA(číslo1;číslo2;...)`.

Číslo1, číslo2, ... je 1 až 255 argumentů, jejichž průměr absolutních odchylek chceme zjistit.

Funkce **PRŮMODCHYLKA** je použita v kapitole 5.

Funkce COVAR

Funkce **COVAR** vypočítá kovarianci, průměr součinů odchylek pro každou dvojici bodů dat. Pomocí kovariance určíme souvislost mezi dvěma soubory dat. Můžeme například zkoumat, zda vyšší příjmy souvisí s vyšším stupněm vzdělání. Zápis funkce:

`=COVAR(pole1;pole2)`.

- **Pole1** je první oblast buněk s čísly.
- **Pole2** je druhá oblast buněk s čísly.

Funkce **COVAR** je použita v kapitole 5.

Funkce CORREL

Funkce **CORREL** vrátí korelační koeficient oblastí buněk **Pole1** a **Pole2**. Pomocí korelačního koeficientu je možné určit vztah mezi dvěma vlastnostmi. Můžeme například zkoumat vztah mezi teplotou určitého místa a používáním klimatizace. Zápis funkce:

```
=CORREL(pole1;pole2).
```

- **Pole1** je první oblast buněk s hodnotami.
- **Pole2** je druhá oblast buněk s hodnotami.

Funkce **CORREL** je použita v kapitole 5.

Funkce LINREGRESE

Funkce **LINREGRESE** vypočítá pomocí metody nejmenších čtverců statistické hodnoty pro přímku, která nejlépe odpovídá uvedeným datům, a vrátí matici s parametry přímky. Vzhledem k tomu, že tato funkce vrací matici hodnot, musí být zadána jako maticový vzorec. Tato příмка je definována následujícím vztahem:

$$y = m \cdot x + b,$$

kde závislé hodnoty y jsou funkcí nezávislých hodnot x . Hodnoty m jsou koeficienty odpovídající každé z hodnot x , b je konstanta.

Zápis funkce:

```
=LINREGRESE(pole_y;[pole_x];[b];[stat]).
```

pole y je povinný argument. Sada hodnot y odvozených ze vztahu $y = m \cdot x + b$.

pole x je nepovinný argument. Sada hodnot x , které již mohou být známy ze vztahu $y = m \cdot x + b$.

b je volitelný argument. Logická hodnota, která určuje, zda se má parametr b (absolutní člen) počítat nebo zda se má rovnat nule.

- Pokud má argument b hodnotu PRAVDA nebo není uveden, počítá se konstanta b běžným způsobem.
- Jestliže má argument b hodnotu NEPRAVDA, uvažuje se, že $b = 0$, a hodnoty m se upraví tak, aby platilo $y = m \cdot x$.

Stat je volitelný argument. Logická hodnota udávající, zda chceme zjistit další regresní statistiky.

- Pokud **stat** je PRAVDA, vrátí funkce **LINREGRESE** další regresní statistiky (viz nápověda k této funkci).
- V případě, že je argument **stat** NEPRAVDA nebo není uveden, vrátí funkce **LINREGRESE** pouze koeficient m a konstantu b .

Funkce **LINREGRESE** je použita v kapitole 5.

Funkce SOUČIN.SKALÁRNÍ

Funkce **SOUČIN.SKALÁRNÍ** vynásobí odpovídající položky uvedených polí (matic) a vrátí součet násobků jednotlivých položek. Například když máme dvě pole 2, 3, 5 a 1, 4, 6, potom funkce **SOUČIN.SKALÁRNÍ** provede výpočet $2 \cdot 1 + 3 \cdot 4 + 5 \cdot 6$. Zápis funkce:

=SOUČIN.SKALÁRNÍ(pole1;pole2;pole3;...).

Pole1, pole2, pole3, ... je 2 až 255 polí (matic), jejichž jednotlivé položky chceme násobit a poté sečíst.



Poznámky:

- Pole uvedená jako argumenty funkce musí být stejně velká. Pokud nejsou, vrátí funkce **SOUČIN.SKALÁRNÍ** chybovou hodnotu **#HODNOTA!**.
- Položky pole, které nejsou číselného typu, zpracovává funkce **SOUČIN.SKALÁRNÍ** jako nuly.

Funkce **SOUČIN.SKALÁRNÍ** je použita v kapitole 6 a 7.

Maticové vzorce

Maticový vzorec nám umožňuje provádět výpočty mezi oblastmi buněk. Například chceme sečíst odpovídající hodnoty v oblastech buněk A1:A5 a B1:B5.

Pomocí *obyčejných vzorců* to provedeme tak, že:

1. Pro součet buňky A1 s buňkou B1 vložíme do buňky C1 vzorec:

$$=A1+B1. \quad (1.4)$$

2. Vzorec zkopírujeme do oblasti buněk C2:C5 a dostaneme vzorec:

$$\begin{aligned} &=A2+B2, \\ &=A3+B3, \\ &=A4+B4, \\ &=A5+B5. \end{aligned} \quad (1.5)$$

Pomocí *maticového vzorce* provedeme výpočet tak, že:

1. Označíme oblast buněk C1:C5.
2. Zapišeme výraz =A1:A5+B1:B5.
3. Stiskneme Ctrl+Shift+Enter.

Do oblasti buněk C1:C5 se zapiše maticový vzorec:

$$\{=A1:A5+B1:B5\}. \quad (1.6)$$



Důležité: Po zapsání výrazu =A1:A5+B1:B5 nesmíme zapomenout stisknout Ctrl+Shift+Enter. Pokud bychom stiskli pouze Enter, maticový vzorec by se nezapsal.



Poznámky:

- Porovnáním vzorců (1.4), (1.5) a (1.6) zjistíme, že při zadání vzorce a jeho kopírování máme v každé buňce (v oblasti buněk C1:C5) jiný vzorec. Každý z těchto vzorců můžeme samostatně upravit nebo vymazat.
- Při zadání maticového vzorce je v každé buňce stejný vzorec.
- Při práci s maticovým vzorcem ve více buňkách je méně pravděpodobné, že vzorec nechtěně přepíšeme. Nemůžeme totiž změnit jedinou buňku maticového vzorce zabírající více buněk.

Maticové vzorce jsou používány v kapitolách 5, 6, a 7.

Ověřování vstupních dat

Před zápisem dat do tabulky lze ověřit, zda splňují určitou podmínku, například jestli jsou ve formátu desetinného čísla a jsou v určitém rozmezí hodnot. Při označení takto ošetřených buněk se zobrazí informativní zpráva o požadavcích na vstupní data a při nesplnění podmínky chybová zpráva. Ověřování vstupních dat ve vybrané oblasti buněk provedeme tak, že:

1. Vyznačíme oblast, ve které chceme data ověřovat.
2. Na kartě **Data** ve skupině **Datové nástroje** stiskneme tlačítko **Ověření dat**.
3. V dialogu **Ověření dat**:
 - Na kartě **Nastavení** určíme podmínku, kterou musí data splnit, aby mohla být do buňky zapsána.
 - Na kartě **Zpráva při zadání** uvedeme zprávu, která se zobrazí, když na buňku umístíme buňkový kurzor.
 - Na kartě **Chybové hlášení** vybereme druh omezení (styl) a doplníme zprávu, která se má zobrazit, není-li podmínka pro zápis data splněna.

Popis některých položek na kartě **Nastavení**:

- **Povolit** – určení typu dat.
- **Rozsah** – nastavení podmínek, které má zapisovaný údaj splnit. U většiny omezení jsou položky minimum a maximum.
- **Přeskakovat prázdné buňky** – prázdné buňky nebudou brány jako chybné.

Ověřování vstupních dat je využíváno v kapitolách 3, 4, 5, 6 a 7.

Podmíněné formátování

Podmíněné formátování slouží ke zvýraznění buněk, ve kterých jsou data, která splňují určité kritérium. Splní-li buňka podmínku, zformátuje se podle podmínky (pravidla), nevyhoví-li podmínce, nezformátuje se.

Podmíněné formátování lze použít na buňku či oblast. Oblast může být i nesouvislá. Buňky upravené podmíněným formátováním lze použít pro řazení a filtrování.

Podmíněné formátování může mít jedno nebo více pravidel. Všechna pravidla se ukládají do dialogu **Správce pravidel podmíněného formátování**. Počet pravidel (podmínek) není omezen a vyhodnocují se podle priorit.

Postup naformátování (přidání pravidla):

1. Vybereme buňku nebo oblast buněk.
2. Na kartě **Domů** ve skupině **Styl** stiskneme tlačítko **Podmíněné formátování**.
3. V seznamu možností vybereme způsob formátování, skupinu podmínek.
4. Ve skupině vybereme pravidlo.
5. Je-li to potřeba, pravidlo upřesníme. To musíme udělat u prvních dvou skupin formátování, kde u některých pravidel musíme zapsat hodnotu a vybrat způsob zvýraznění. Zvýraznění můžeme také sami naformátovat v dialogu **Formát buněk** po výběru položky **Vlastní formát**.

Podmíněné formátování je využíváno v kapitole 3, 4 a 6.

Nástroje pro analýzu dat

Součástí Excelu jsou nástroje pro analýzu dat, které pomohou při výpočtu proměnné (proměnných) pro dosažení požadovaného výsledku. Pro techniky jsou důležité zejména tyto nástroje pro analýzu dat:

- Hledání řešení,
- Řešitel,
- Scénáře,
- Kovariance,
- Korelace,
- Regrese.

K těmto základním nástrojům pak můžeme přidat celou řadu dalších.

V této kapitole se zaměříme na nástroje pro analýzu dat, které budou využity v této knize.

Hledání řešení

Nástroj **Hledání řešení** použijeme v situaci, kdy máme vzorec (funkci) a hledáme hodnotu jedné proměnné pro dosažení žádaného výsledku.

Postup si ukážeme na jednoduchém příkladě, kdy máme v buňce C1 vzorec pro sečtení buněk A1 a B1, ve kterých jsou hodnoty 45 a 57 (výsledek součtu je 102):

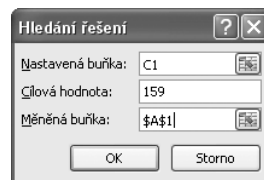
=A1+B1. (1.7)

Chceme zjistit, jakou hodnotu musí mít buňka A1, aby výsledek součtu byl 159. Budeme postupovat tak, že:

1. Zvolíme kartu **Data**.
2. Klepneme ve skupině **Datové nástroje** na položku **Analýza hypotéz**.
3. Zvolíme **Hledání řešení**.
4. V dialogu **Hledání řešení** (viz obrázek 1.6):
 - v okně **Nastavená buňka** vytyčením zadáme buňku C1.
 - v okně **Cílová hodnota** zapíšeme číslo 159.
 - v okně **Měněná buňka** vytyčením zadáme buňku A1.

Po potvrzení zadaných údajů dostaneme v buňce C1 požadovanou hodnotu 159 a v buňce A1 zjištěnou hodnotu 102.

Nástroj **Hledání řešení** je využíván v kapitole 3.



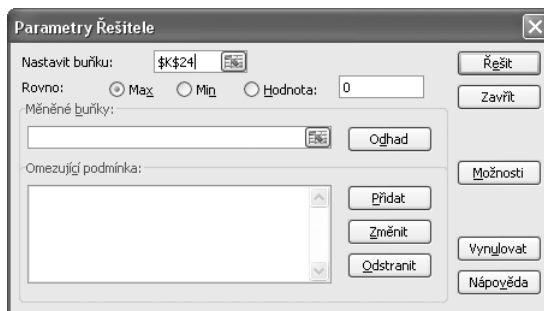
Obrázek 1.6 Dialog Hledání řešení

Řešitel

V této knize budeme nástroj **Řešitel** používat pro nalezení minimální nebo maximální hodnoty u matematických modelů, které se skládají z jednoho nebo více vzorců (hledání optimálního řešení).

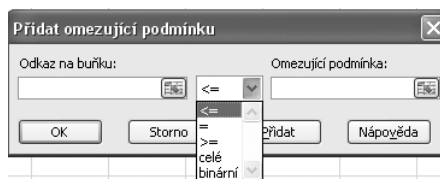
Postup použití **Řešitele**:

1. Na listu sešitu vytvoříme matematický model, který chceme řešit.
2. Zadáme příkaz **Data → Řešitel**.
3. V dialogu **Parametry Řešitele** (viz obrázek 1.7):
 - V okně **Nastavit buňku** vytyčným zadáme adresu cílové buňky, pro kterou hledáme řešení.
 - V nabídce **Rovno** volíme **Max** nebo **Min** (maximalizace nebo minimalizace hodnoty cílové buňky).
 - Vytyčením zadáme **Měněné buňky**, tj. buňky, do nichž bude spočítán výsledek matematického modelu.



Obrázek 1.7 Dialog Parametry Řešitele

Omezující podmínky vytvoříme klepnutím na tlačítko **Přidat** (zobrazí se dialog **Přidat omezující podmínku**, viz obrázek 1.8).



Obrázek 1.8 Dialog Přidat omezující podmínku

1. Do oken **Odkaz na buňku** a **Omezující podmínka** zadáme adresy buněk, mezi kterými vytváříme požadované vztahy (relace).
2. Z nabídky uprostřed vybereme příslušný relační operátor (např. $\leq$).
3. Nastavení potvrdíme stiskem tlačítka **OK**.
4. V dialogu **Parametry Řešitele** (viz obrázek 1.7) klepneme na tlačítko **Možnosti**.
5. V dialogu **Možnosti Řešitele** nastavíme požadované parametry.

Řešení naší úlohy dostaneme klepnutím na tlačítko **Řešit** v dialogu **Parametry Řešitele**.

V dialogu **Možnosti Řešitele** (viz obrázek 1.9) lze upřesnit způsob řešení optimalizační úlohy. Význam polí a tlačítek:

Maximální doba: Omezí dobu trvání procesu řešení. Do tohoto textového pole můžeme zadat hodnotu až 32 767, výchozí hodnota 100 (sekund) je však dostatečně dlouhá pro řešení většiny menších problémů.

Iterace: Omezí dobu trvání výpočtu pomocí omezení počtu předběžných výpočtů. Do tohoto textového pole můžeme zadat hodnotu až 32 767, výchozí hodnota 100 je však dostatečně dlouhá pro řešení většiny menších problémů.

Přesnost: Ovládá přesnost řešení pomocí zadaného čísla určujícího, zda hodnota buňky s omezující podmínkou odpovídá požadované hodnotě nebo zda nepřesahuje horní či dolní mez. Přesnost musí být zadána jako desetinné číslo v rozmezí 0 (nula) až 1. Vyšší přesnosti můžeme dosáhnout, zadáme-li číslo s větším počtem desetinných míst. Hodnota 0,0001 označuje například vyšší přesnost než hodnota 0,01.

Tolerance: Označuje procentuální hodnotu, o kterou se může lišit vypočítaná hodnota cílové buňky splňující celočíselnou omezující podmínku od skutečné optimální hodnoty tak, aby

byla ještě považována za přijatelnou. Tato možnost se používá pouze u problémů s celočíselnými omezujícími podmínkami. Vyšší hodnota tolerance urychluje proces řešení.

Konvergence: Pokud je relativní změna hodnoty cílové buňky u pěti posledních iterací nižší než číslo zadané v textovém poli **Konvergence**, ukončí **Řešitel** výpočet. Konvergence se používá pouze u nelineárních problémů a musí být zadána jako desetinné číslo v rozmezí 0 (nula) až 1. Menší konvergenci získáme, zadáme-li číslo s větším počtem desetinných míst. Hodnota 0,0001 označuje například menší relativní změnu než hodnota 0,01. Čím je hodnota konvergence nižší, tím déle trvá v **Řešiteli** proces řešení.

Lineární model: Zaškrtnutí políčka urychlí proces řešení v případě, že jsou všechny vztahy v modelu lineární a chceme vyřešit lineární optimalizační problém.

Nezáporná čísla: Zaškrtnutí políčka způsobí, že v **Řešiteli** bude platit dolní mez 0 (nula) pro všechny měnitelné buňky, u kterých jsme do textového pole **Omezující podmínka** v dialogovém okně **Přidat omezující podmínku** nezadali hodnotu dolní meze.

Automatické měřítko: Zaškrtnutí políčka aktivuje automatickou úpravu měřítka v případech, kdy se výrazně liší velikost vstupů a výstupů, například při maximalizaci procenta zisku podle investic v milionech korun.

Zobrazit výsledek iterace: Zaškrtnutí políčka způsobí, že po každé iteraci přeruší **Řešitel** výpočet a zobrazí výsledek iterace.

Extrapolace.

Určí metodu, která bude použita k získání počátečního odhadu základních proměnných v každém jednorozměrném vyhledávání.

Lineární: Použije lineární extrapolaci tangenciálního vektoru.

Kvadratická: Použije kvadratickou extrapolaci, která může zlepšit výsledky u vysoce nelineárních problémů.

Derivace:

Určí metodu stanovení rozdílů, která bude použita při odhadu parciálních derivací funkcí cíle a omezující podmínky.

Standardní: Používá se u většiny problémů, u kterých se hodnoty omezení mění poměrně pomalu.

Přesná: Používá se u problémů, u kterých se omezující podmínky mění rychle, obzvláště v okolí mezí. Tato možnost vyžaduje více výpočtů, může však být užitečná v případech, kdy **Řešitel** zobrazí zprávu, že nelze nalézt vhodnější řešení.

Metoda:

Určí algoritmus, který bude použit v jednotlivých iteracích k určení směru vyhledávání.

Newtonova: Použije kvazi-Newtonovu metodu, která obvykle vyžaduje více paměti, avšak menší počet iterací než sdružená gradientní metoda.

Sdružená: Vyžaduje menší kapacitu paměti než Newtonova metoda, ale k dosažení určité úrovně přesnosti obvykle potřebuje větší počet iterací. Tuto možnost můžeme použít při řešení rozsáhlého problému v případě, že máme k dispozici omezenou kapacitu paměti, nebo v případě, že je procházení iteracemi pomalé.

Načíst model. Zobrazí dialog **Načíst model**, ve kterém můžeme zadat odkaz na model, který chceme načíst.

Uložit model. Zobrazí dialog **Uložit model**, ve kterém můžeme zadat umístění, do kterého chceme model uložit. Toto tlačítko použijeme pouze v případě, že chceme s listem uložit více než jeden model. První model bude uložen automaticky.

Nástroj **Řešitel** je využíván v kapitole 5, 6 a 7.

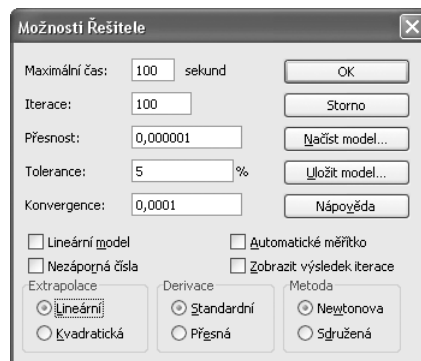
Scénáře

Scénář je pojmenovaná množina (soubor) hodnot buněk. Každá množina hodnot je jeden scénář. Výběrem scénáře se změní hodnoty v buňkách. Scénáře jsou nástrojem pro simulaci různých stavů. Umožňují v jedné tabulce zobrazit různé kombinace dat, a tak nahradit mnoho tabulek.

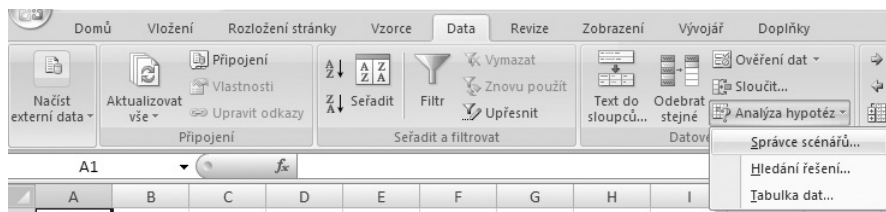
- Scénáře má význam vytvářet pro buňky, ve kterých se mění hodnoty. Scénář obsahuje hodnoty, ne formáty. Nelze jej zkopírovat, musí se vytvořit znovu.
- Jeden scénář může mít až 32 měněných buněk. Více buněk rozdělíme do více scénářů.
- Oblast měněných buněk musí být na jednom listu, nemusí tvořit souvislou oblast a může být pojmenovaná. Oblasti se mohou překrývat.
- Měněná buňka může obsahovat jen konstantu. Vzorec bude přepsán výsledkem.

Postup vytvoření scénáře:

1. Zvolíme kartu **Data**.
2. Klepneme ve skupině **Datové nástroje** na položku **Analýza hypotéz**.
3. Volíme **Správce scénářů** (viz obrázek 1.10).
4. V dialogu **Správce scénářů** (viz obrázek 1.13) klepneme na tlačítko **Přidat** a zobrazí se dialog **Přidat scénář** (viz obrázek 1.11).
5. Do textového pole **Název scénáře** zadáme název scénáře (například Varianta 1).
6. Do textového pole **Měněné buňky** zadáme odkazy na buňky, které chceme měnit (například A1 a A2).
7. Do textového pole **Komentář** zadáme libovolný komentář.



Obrázek 1.9 Dialog Možnosti Řešitele

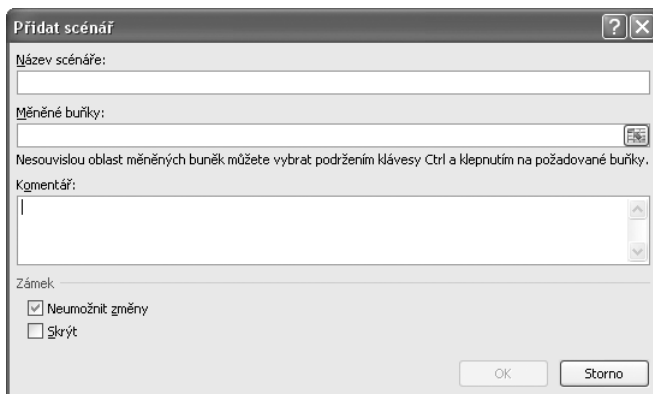


Obrázek 1.10 Spuštění Správce scénářů

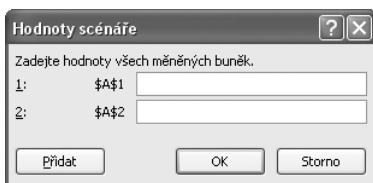
Po potvrzení zadaných údajů se zobrazí dialog **Hodnoty scénáře** (viz obrázek 1.12), ve kterém jsou zobrazeny původní hodnoty z tabulky. Tyto hodnoty můžeme změnit.

Odsouhlasením údajů se dostaneme zpět na dialog **Správce scénářů** (viz obrázek 1.13), ve kterém už máme scénář **Varianta 1**. Takovýchto scénářů (s různými hodnotami měněných buněk) si můžeme vytvořit libovolné množství. Z nabídky pak

vybereme scénář, který chceme použít pro výpočet (např. Varianta 1), a klepneme na tlačítko **Zobrazit**. Do buněk A1, a A2 se zapíše nové hodnoty ze scénáře.



Obrázek 1.11 Dialog Přidat scénář



Obrázek 1.12 Dialog Hodnoty scénáře

Popis dialogu **Správce scénářů** (viz obrázek 1.13):

Scénáře: Seznam všech vytvořených scénářů na listu.

Měněné buňky: Oblast měněných buněk pro označený scénář.

Komentář: Komentář doplněný programem a zapsaný uživatelem.

Přidat: Zobrazení dialogu **Přidat scénář**.

Odstranit: Odstranění označeného scénáře. Scénář už nelze vrátit!

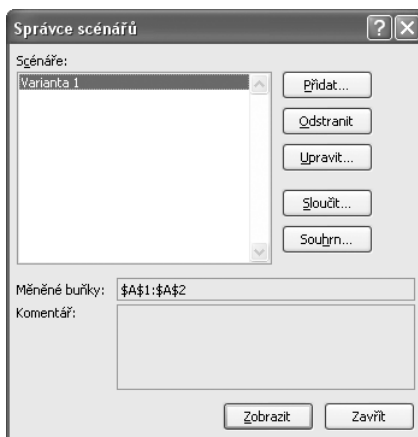
Upravit: Zobrazení dialogu **Upravit scénář**. Slouží ke změně scénáře (i jeho názvu). Dialogy **Přidat scénář** a **Upravit scénář** mají stejný obsah.

Sloučit: Zobrazí se dialog **Sloučit scénáře** pro přidání scénářů z jiného listu sešitu. Má použití jen u listů se stejnými oblastmi měněných buněk.

Souhrn: Zobrazí se dialog **Zpráva scénáře** pro výběr výstupu: **Zpráva scénáře** nebo **Kontingenční tabulka** a oblasti výsledných buněk, pro kterou se mají zprávy vytvořit. Slouží pro další rozbor. Výsledné buňky mohou být nespojitou oblastí.

Zobrazit: Hodnoty z označeného scénáře se promítnou do měněných buněk.

Scénáře jsou využívány v kapitolách 3, 6 a 7.



Obrázek 1.13 Dialog Správce scénářů

Kovariance

Nástroj **Kovariance** vypočítá hodnotu funkce **COVAR** pro jednotlivé dvojice měřených proměnných. (Přímé použití funkce **COVAR** místo nástroje **Kovariance** je vhodnou alternativou v případě, že existují pouze dvě měřené proměnné, tj. $N = 2$.) Položka na diagonále výstupní tabulky nástroje **Kovariance** v řádce i a sloupci i je kovariance i -té měřené proměnné samé se sebou; jedná se o rozptyl základního souboru u dané proměnné, který je vypočítán pomocí funkce **VAR**.

Nástroj **Kovariance** můžeme použít k testování jednotlivých dvojic měřených proměnných a zjištění závislosti dvou měřených proměnných. Závislost znamená, že vysoké hodnoty jedné proměnné odpovídají vysokým hodnotám druhé proměnné (kladná kovariance) nebo že nízké hodnoty jedné proměnné odpovídají vysokým hodnotám druhé proměnné (záporná kovariance). Pokud jsou hodnoty obou proměnných nezávislé, bude kovariance blízká nule.

Nástroj **Kovariance** je využíván v kapitole 5.

Korelace

Nástroj **Korelace** můžeme použít k testování jednotlivých dvojic měřených proměnných a zjištění závislosti dvou měřených proměnných. Závislost znamená, že vysoké hodnoty jedné proměnné odpovídají vysokým hodnotám druhé proměnné (kladná korelace) nebo že nízké hodnoty jedné proměnné odpovídají vysokým hodnotám druhé proměnné (záporná korelace). Pokud jsou hodnoty obou proměnných nezávislé, bude korelace blízká nule.

Korelační koeficient podobně jako kovariance udává, do jaké míry se vzájemně mění dvě měřené proměnné. Na rozdíl od kovariance je u korelačního koeficientu použito měřítko; jeho hodnota tedy nezávisí na jednotkách, v nichž jsou dané dvě měřené proměnné vyjádřeny. (Jsou-li dvě měřené proměnné například hmotnost a výška, hodnota korelačního koeficientu se při převodu liber na kilogramy nezmění.) Korelační koeficient musí mít hodnotu v rozsahu od -1 do +1 (včetně).

Nástroj **Korelace** je využíván v kapitole 5.

Regrese

Nástroj **Regrese** provede lineární regresi tak, že pomocí metody nejmenších čtverců proloží přímkou sadou pozorování. Regrese umožňuje analyzovat, jakým způsobem ovlivňují hodnoty jedné nebo více nezávislých proměnných hodnotu jedné závislé proměnné. Můžeme například provést analýzu vlivu věku, hmotnosti a výšky na výkon sportovce. Na základě skupiny výsledků můžeme každému z těchto tří faktorů přiřadit podíl na výkonu a pomocí získaných hodnot předpovědět výkon nového netestovaného sportovce.

Nástroj **Regrese** používá funkci **LINREGRESE** a je využíván v kapitole 5.

Jednotný vzhled sešitu a práce se seznamy

K dosažení jednotného vzhledu sešitu všech dokumentů vytvořených v aplikacích 2007 slouží motivy. Motiv je kombinace barev, znakových sad a efektů, která neobsahuje data. Každý

sešit je vytvořen na základě motivu a veškeré objekty v sešitu přebírají grafiku motivu. Přiřazením motivu snadno a rychle naformátujeme celý sešit. Pokud vybereme jiný motiv, změní se grafické provedení celého sešitu. Je připraveno 20 motivů, společných pro aplikace Office 2007. Motivy lze upravovat a přidat vlastní. Další motivy lze stáhnout z webu, ze služby Microsoft Office Online.

Motivy

Kombinace barev, znakových sad a efektů zvoleného motivu se promítá do galerie barev u písma, výplně buněk, ohraničujících čar, rychlých stylů, stylů buněk a tabulek, grafů, objektů atd. (do veškeré grafiky sešitu).

Použití motivu

Galerii motivů najdeme na kartě **Rozložení stránky** ve skupinovém rámečku **Motivy** u tlačítka **Motivy** (viz obrázek 1.14). V nabídce je 20 motivů. Každý nový sešit je založen na výchozím motivu a ten se použije na všechny listy sešitu. Výchozím motivem je kancelář, která používá pro základní text písmo Calibri, nadpisy Cambria. Základní velikost písma je určena v nastavení pracovního prostředí, v dialogu **Možnosti aplikace Excel** v kategorii **Oblíbené**.

Veškeré grafické provedení sešitu je dynamicky propojeno s motivem, takže změnou motivu se automaticky změní vzhled sešitu. Přemísťováním ukazovátkem myši přes galerii motivů se vybraný motiv v objektech vizualizuje. Klepnutím myši motiv akceptujeme, klepnutím mimo galerii nebo klávesou **Esc** zůstane původní grafické provedení.

Můžeme použít předdefinované motivy sešitu nebo vytvořit vlastní motivy úpravou komponenty (barvy, znakové sady a efekty) existujícího motivu a ten uložit jako vlastní motiv.



Obrázek 1.14 Dialog Motivy

Barvy motivů

Barvy motivů zahrnují 12 barev, dvě základní barvy textu a pozadí: černá a bílá. Další barvy se mění podle motivu: dvě doplňkové barvy jsou pro text a pozadí. Dvojice barva textu a pozadí je určena pro tmavý text na světlém pozadí nebo světlý text na tmavém pozadí.

V každé galerii barev jsou odstíny barev založeny na zvoleném motivu. V galerii barev je v části **Standardní barvy** deset stálých barev bez ohledu na použitý motiv.

Barvy přizpůsobíme tak, že:

1. Zvolíme kartu **Rozložení stránky**.
2. Klepneme ve skupině **Motivy** na položku **Motivy**.
3. Zvolíme motiv, který bude předlohou pro barvy (viz obrázek 1.14).