

## Tabulkový kalkulátor

Tabulkový procesor (anglicky spreadsheet) je program zpracovávající tabulku informací (je to vlastně matice). V jednotlivých buňkách mohou být uložena data či vzorce počítající s těmi daty. V tom případě se v tabulce zobrazují data vypočtená ze vzorců. Dnes jsou hojně integrovány do kancelářských balíčků. Zprvu byl tabulkový procesor využíván zejména ve finančnictví, proto byly první verze vybaveny zejména funkcemi vhodnými na finanční výpočty, dnes ho však jde využít k širokému množství výpočtů a jiných zpracování dat.

V současné době je nejpoužívanější Microsoft Excel pro Microsoft Windows a Mac OS[zdroj?]. Na ostatních operačních systémech je asi OpenOffice.org Calc, který proniká i na Microsoft Windows.

Když programátorské jazyky umožnily tvoření tisknutelných sestav, vedlo to ke snaze o efektivnější organizaci dat. První tabulkové procesory data přepočítávaly v pořadí určeném jejich pozicí v tabulce, ale později se přešlo na reprezentaci v grafu závislosti, kde jednotlivé obsazené buňky tvoří uzly a hrany jsou odkazy na jiné buňky zmíněné ve vzorci. Takto se při změně hodnoty buňky mohou přepočítávat jen buňky změnou zasažené.

Síla tabulkových procesorů spočívá v tom, že lidé považují za lehčí zorientovat se v prostorových vazbách než psát obdobný program pro zpracování po krocích.

Myšlenka elektronického zpracování tabulek byla poprvé načrtnuta v „Budgeting Models and System Simulation“ od Richarda Mattessicha. Samotný předobraz tabulkových procesorů je ale mnohem starší, nalezneme jej v účetnictví (viz Historie účetnictví). Část zásluhy na vzniku tabulkových procesorů lze přičíst i pánům jména Pardo a Landau s programem LANPAR, kteří podali patent (U. S. Patent číslo 4 398 249) na některé algoritmy v r. 1970. Patent byl odmítnut, ale následná žaloba byla uznána, čímž byl dán základ k softwarovým patentům. Samotný vynález tabulkového procesoru ale náleží Danu Bricklinovi. Legenda říká, že Bricklin sledoval svého univerzitního profesora na Harvard Business School dělat tabulku výsledků výpočtu na tabuli. Když chtěl profesor opravit chybu nebo změnit parametr, musel nudně mazat a přepisovat množství následných záznamů v tabulce. Bricklin to dovedlo k myšlence replikovat proces na počítači užívajíc hodnot z tabule k zobrazování výsledků níže postavených vzorců. Proto napsal v r. 1979 VisiCalc.

V první polovině 80. let byl nejrozšířenějším tabulkovým procesorem Lotus 1-2-3. Z mnoha jeho konkurentů se jako nejúspěšnější ukázal Microsoft Excel. Ten slavil úspěchy se svým grafickým uživatelským rozhraním a využitím myši nejprve na počítačích Macintosh (od r. 1985), pak i na IBM PC s Microsoft Windows.

Od počátku 21. století je kladen mnohem větší důraz na vzhled a zlepšování uživatelského rozhraní. Tabulkové procesory začínají měnit svou podobu (Excel 2002 > 2003 > 2007; IBM Lotus Symphony Spreadsheets; OpenOffice.org Calc 1.x > 2.x > 3.x; ...) nejen změnou ikon a barevnými přechody, v některých případech dochází k použití nového rozvržení panelů (Excel 2007; Corel Home Office Calculate 2010;...) označovaného pojmem Ribbon.


## Základní funkce

Každý tabulkový kalkulátor obsahuje celou řadu předdefinovaných funkcí. Z těch jednoduchých si můžeme představit například tyto:

### *suma(argument\_1; argument\_2; ... argument\_n)*

	A
1	1
2	2
3	3
4	=SUMA(A1:A3)

Tato funkce provede součet argumentů. Argumenty mohou být buď číselné hodnoty nebo odkazy na buňky s hodnotami, které mají být sečteny. Argumenty mohou tvořit i skupinu sousedních buněk, tzv. interval, který se zapíše s dvojtečkou, čili ve tvaru *argument\_1:argument\_n*.

Tuto funkci můžete vložit též tlačítkem AutoSum, které se nachází  ve standardním panelu nástrojů. Další alternativou je vložení klávesovou zkratkou (Alt + =) v buňce, kde má být součet proveden.

### *průměr(argument\_1; argument\_2; ... argument\_n)*

	A
1	1
2	2
3	3
4	=PRŮMĚR(A1:A3)

S touto funkcí zjistí tabulkový kalkulátor aritmetický průměr ze zadaných hodnot, které jsou argumenty této funkce. S argumenty se pracuje stejným způsobem jako u předchozí funkce.

### *dnes()*

	A
1	=DNES()

Funkce *dnes()* vloží do buňky vždy aktuální datum. Tato funkce nemá žádný argument.

### *když(podmínka; ano; ne)*

	A	B
1	0	1
2	dívka	chlapec
3	=KDYŽ(A1=1;"chlapec"; "dívka")	=KDYŽ(B1=1;"chlapec"; "dívka")

Tato logická funkce pracuje s podmínkou. Podmínka je prvním argumentem této funkce. Po vyhodnocení funkce rozhodne, zda použije druhý argument, který se

provede v okamžiku, kdy podmínka splněna je, nebo třetí argument, který se provede v okamžiku, kdy podmínka splněna není.

### *velká(argument)*

	A
1	tabulkový kalkulátor
2	TABULKOVÝ KALKULÁTOR
3	=VELKÁ(A1)

Textová funkce *velká()* převádí řetězec, který je argumentem této funkce na velká písmena. Argumentem může být buď přímo řetězec, ale častěji se používá odkazu na buňky s textem.

### *countif(argument\_1; argument\_2)*

	A	B	C	D	E	F
1	1	1	0	1	0	1
2	4	=COUNTIF(A1:F1;1)				

Statistická funkce *countif()* vrací počet buněk, které splňují zadané kritérium. První argument označuje oblast, která se bude procházet a druhý argument pak obsahuje zadané kritérium.