

Seminář IIT

MS Excel,
opakování funkcí

Opuka

- Opakování z minulé hodiny.
 - Založeno na výsledcích Vašich domácích úkolů, podrobné zopakování věcí, ve kterých děláte nejčastěji chyby.
 - Nejčastější jsou následující dva typy chyb (typicky v kombinaci):
 - Nepochopení principu, jak obecně fungují v MS Excelu funkce jako takové (toto je spíše o syntaxi).
 - Místo toho, abyste se učili používat jednotlivé funkce, pochopte princip, na jakém fungují všechny!!
 - Nezvládnutí logických funkcí (po sémantické stránce).
 - Podmínky, logické operátory,
- Většinu hodiny budou tvořit cvičení.

Písemná práce

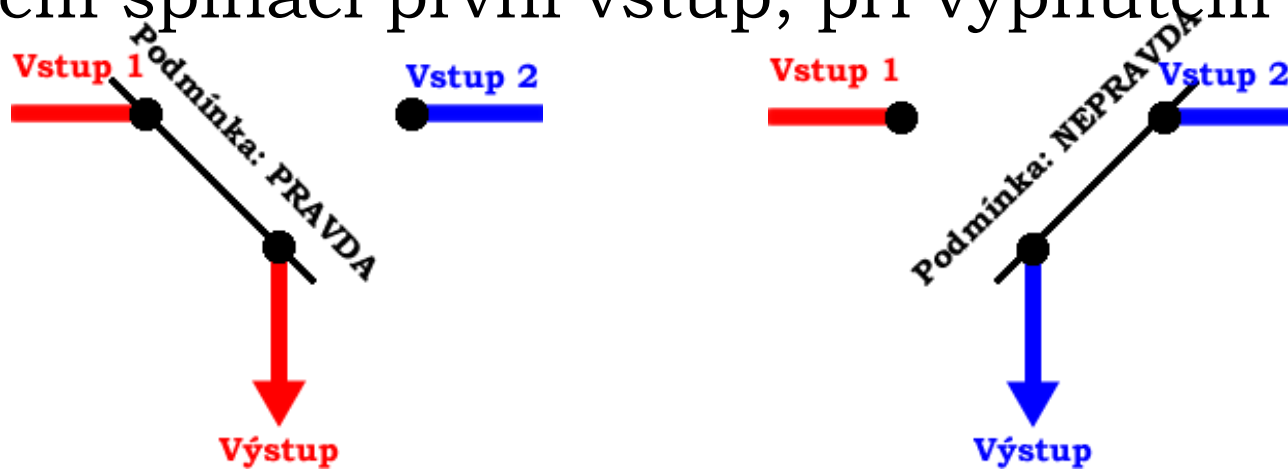
- **!! PŘÍŠTÍ HODINU SE PÍŠE PÍSEMNÁ PRÁCE !!**
- Bude zadán jediný příklad, na který budete mít hodně času (minimálně dvě vyučovací hodiny).
- Budete mít povoleno používat libovolné podklady, své práce z předchozích hodin (i této), Google, manuály MS Excelu, knihy, skripta, tutoriály,
 - Zakázána bude pouze komunikace s libovolnými entitami ve vesmíru, zejména pak s lidmi ;o)
- **Písemka** nebude těžká, ale **bude vyžadovat Vaše POCHOPENÍ funkcí (zejména logických).**
 - Pochopení **NEZNAMENÁ** umět udělat cosi podle vzoru, který lehce modifikujete, nýbrž aktivní porozumění.

Funkce IF / KDYŽ

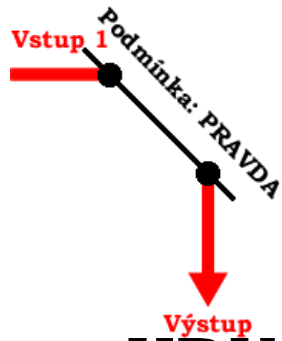
- Jedna z nejsilnějších funkcí MS Excelu, je třeba ji **PLNĚ CHÁPAT**, nikoliv občas správně natipovat.
 - Budete-li mít problém vůbec tu funkci použít a syntakticky zapsat, těžko lze počítat s tím, že budete pomocí té funkce něco složitějšího řešit.
- Funkce má tři parametry. Prvním parametrem je logický výraz, druhé dva parametry jsou výrazy libovolného typu.
 - Třetí parametr je sice nepovinný, ale uvádějte ho, implicitně je nastavený na hodnotu NEPRAVDA.
- Výstup funkce je výraz, je tedy možno jej použít jako parametr jiné funkce KDYŽ (je možné je vnořovat).

Sémantika funkce $I\mathcal{F} / \mathcal{K}\mathcal{D}\mathcal{V}\mathcal{Z}$

- Když je potřeba, aby buňka zobrazovala například výstupy dvou různých funkcí v závislosti na vnější podmínce, řeší se to právě touto funkcí.
- První parametr funkce je vždy logický výraz, který tedy může nabývat pouze dvou hodnot.
- Funkci si lze představit jako přepínač, který má dva vstupy a jeden výstup, přičemž na výstup posílá při zapnutém spínači první vstup, při vypnutém druhý.



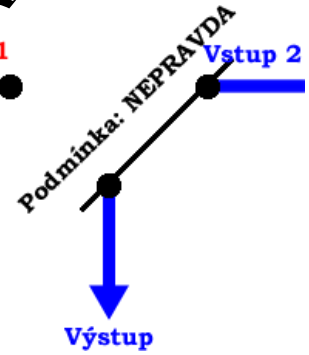
Syntax a příklady funkce IF / KDYŽ



Vstup 2

=KDYŽ(podmínka; vstup 1; vstup 2)

Vstup 1



● =KDYŽ(PRAVDA; "ANO"; "NE")

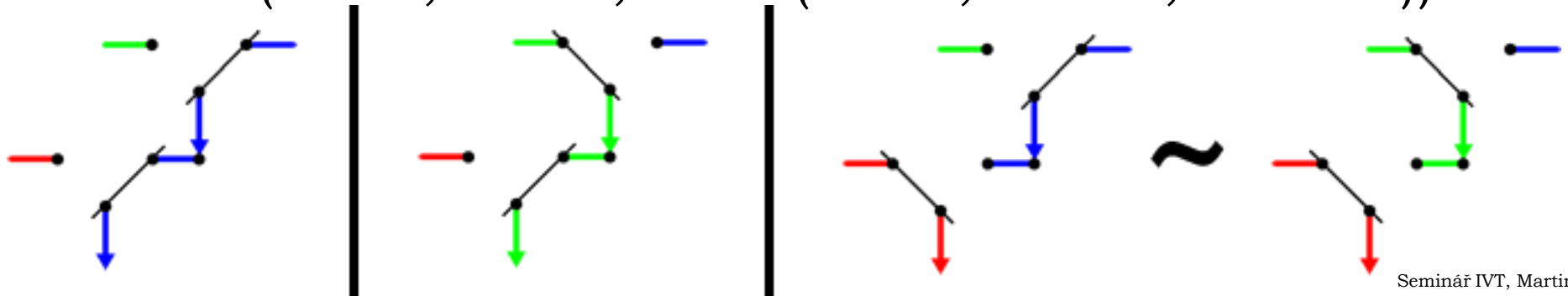
● =KDYŽ(NEPRAVDA; "NEPRAVDA"; "PRAVDA")

● **"NEPRAVDA" a NEPRAVDA jsou různé věci!!**

● =KDYŽ(0<2;"8";"0") vs. =KDYŽ(0<2;8;0)

● Jednou je výsledkem řetězec, podruhé číslo.

● =KDYŽ(2=A1;"DVA";KDYŽ(2<A1;"MOC";"MALO"))



Cvičení

- Napište funkci, která se bude chovat přesně stejně jako funkce `signum` (samozřejmě bez použití funkce `SIGN`), jako parametr přijme fixně nějakou buňku.
- Napište funkci, která počítá zbytek po dělení hodnot ve dvou buňkách a ošetřuje dělení nulou (tedy kdyby mělo k dělení nulou dojít, vypíše se chybová hláška, kterou si pro ten účel vymyslíte – nikoliv `#DIV/0`).
 - Funkci zbytku si někde najděte (nejlépe asi Google).
- Do buněk A1 a C1 budou uživatelé zadávat čísla, do buňky B1 operaci (jeden ze znaků `+` `-` `*` `/`). Vámi vytvořená funkce bude vracet výsledek (tedy čísla A1 a C1 mezi sebou sečte/odečte/vynásobí/vydělí).

Logické výrazy

- Jedná se o výrazy, jejichž výsledkem je právě jedna z hodnot NEPRAVDA/PRAVDA (0/1, NE/ANO, ...).
- Atomické logické výrazy (neobsahují operátory):
 - $2 < 3$, $2 = 4$, venku prší, nastala chyba, $B52 > F16$,
- Složené logické výrazy (atomické výrazy s operátory).
 - Logické operátory: AND, OR, NOT, XOR, IMP.
 - Každý z nich (krom NOT) lze ekvivalentně zapsat pomocí NOT a libovolného jiného operátoru.
 - U implikace záleží na pořadí operátorů (v IT nevhodné).
 - MS Excel umí AND, OR a NOT (syntax A, NEBO, NE).
 - XOR z nějakého důvodu chybí (sám nevím proč), ale je možné ho samozřejmě nahradit ostatními operátory.

Logické operátory nejsou nic tajemného

- Logické operátory nejsou nic tajemně mystického, ač většina z Vás na to při výuce programování tak hleděla.
 - Logické operátory jsou stejné jako kterékoliv jiné operátory, například klasické početní.
 - Podobně jako $1 + 1 = 2$, tak například $1 \text{ AND } 1 = 1$.
 - Logické operace probíhají nad binární abecedou (tedy nad logickými hodnotami ne/ano, můžeme je označit jako 0/1).
 - Není v nich žádná hluboká myšlenka. Podobně, jako když se učíte v první třídě základní školy sčítat čísla, můžete se naučit čísla ANDovat či ORovat.
 - ANDování či ORování je rozhodně lehčí než sčítání.

Co logické operátory znamenají?

- NOT je negace, znamená něco jako opak.
 - Znamená to de facto totéž, s čím se setkáme v běžném životě (když například někdo neguje vše, co řeknete).
- AND je konjunkce, znamená „a zároveň“.
 - Je splněna v případě, že jsou splněny všechny podmínky (vezměte si občanský průkaz A pas).
- OR je disjunkce, znamená „nebo“.
 - K jejímu splnění stačí, aby byla splněna alespoň jedna podmínka (vezměte si občanský průkaz NEBO pas).
- XOR je ostrá disjunkce, znamená „buď - a nebo“.
 - Splněná musí být právě jedna podmínka (Být, či nebýt, to je, oč tu běží. -Hamlet).

Obecně o vyhodnocování logických operátorů

- Operátor NOT je unární.
 - $\text{NOT}(0) = 1$ a $\text{NOT}(1) = 0$.
- Operátor AND je binární.
 - $0 \text{ AND } 0 = 0$, $0 \text{ AND } 1 = 0$, $1 \text{ AND } 0 = 0$, $1 \text{ AND } 1 = 1$.
- Operátor OR je binární.
 - $0 \text{ OR } 0 = 0$, $0 \text{ OR } 1 = 1$, $1 \text{ OR } 0 = 1$, $1 \text{ OR } 1 = 1$.
- Operátor XOR je binární.
 - $0 \text{ XOR } 0 = 0$, $0 \text{ XOR } 1 = 1$, $1 \text{ XOR } 0 = 1$, $1 \text{ XOR } 1 = 0$.
- Operátor IMP je binární a zapisuje se šipkou \rightarrow .
 - $0 \rightarrow 0 = 1$, $0 \rightarrow 1 = 1$, $1 \rightarrow 0 = 0$, $1 \rightarrow 1 = 1$.
 - Z čehož je evidentní, že záleží na pořadí operandů.

Tabulky vyhodnocování operátorů

- Pro lepší pochopení, aby bylo jasné, že logické operátory nejsou nic mystického, uvádím i tabulky obyčejného sčítání a násobení.
 - Odčítání a dělení neuvádím jen proto, že v nich záleží na pořadí operátorů (u AND, OR a XOR také nezáleží).

+	0	1	2	3
0	0	1	2	3
1	1	2	3	4
2	2	3	4	5
3	3	4	5	6

*	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	4	6
3	0	3	6	9

AND	0	1
0	0	0
1	0	1

OR	0	1
0	0	1
1	1	1

XOR	0	1
0	0	1
1	1	0

- Dvourozměrnou (tedy se řádky a sloupci) tabulku pro NOT nemá smysl dělat, protože NOT je unární operátor, ne binární (má tedy jen jeden argument).
 - Not funguje prostě tak, že z nuly dělá jedničku a z jedničky nulu. $\text{NOT}(0) = 1$ a $\text{NOT}(1) = 0$.

Říklady v praxi

- Atomický logický výraz nabývá hodnot 0/1 (to se zjistí snadno podle toho, jestli není/je pravdivý).
- Složený logický výraz se vyhodnotí tak, že se aplikují operátory na atomické výrazy, které složitý výraz obsahuje.
- „Žijeme v ČR a zároveň je rok 2377.“ →
„Žijeme v ČR.“ AND „Je rok 2377.“ → 1 AND 0 → 0
 - První úpravou jsme slovní spojení „a zároveň“ převedli na AND.
 - Druhou úpravou jsme dosadili 1 za pravdivá tvrzení a 0 za nepravdivá (jen syntaktický zápis).
 - Nakonec jsme vyhodnotili 1 AND 0 (podle tabulek).

Logické operátory v MS Excelu

- V MS Excelu jsou tři logické operátory AND, OR a NOT (implementovány jako funkce A, NEBO a NE).
- Funkce NE je unární (má jediný argument).
- Funkce A a NEBO jsou n-ární (tedy mají libovolný počet argumentů), což trochu mění způsob jejich vyhodnocování oproti klasickým binárním.
 - Funkce A vrací PRAVDA, pokud jsou všechny její argumenty PRAVDA. Je-li alespoň jeden NEPRAVDA, pak je výsledkem NEPRAVDA.
 - Funkce NEBO vrací PRAVDA, pokud je alespoň jeden její argument PRAVDA. Jsou-li všechny argumenty NEPRAVDA, pak je výsledkem NEPRAVDA.

Cvičení

- Pouze pomocí funkce KDYŽ napište funkci, která provádí operaci AND nad buňkami A1 a B1. Tedy by měla dělat totéž, co $=A(A1, B1)$.
- Po vzoru prvního zadání napište:
 - OR pouze pomocí KDYŽ.
 - XOR pouze pomocí KDYŽ.
 - AND pouze pomocí NOT a OR (napřed si rozmyslete).
 - OR pouze pomocí NOT a AND (napřed si rozmyslete).
- Pro účely testování svých funkcí pamatujte na to, že hodnoty PRAVDA/NEPRAVDA jsou normální funkce, tedy pokud chcete do A1 zadat hodnotu PRAVDA, napište do A1 prostě $=PRAVDA$.

Funkce pro písemnou práci (bude se psát příště)

- Mohou se Vám hodit některé z následujících funkcí:
 - Z logických úplně všechny, neopomeňte IFERROR!!
 - Z textových: DÉLKA, CONCATENATE, ČÁST, DOSADIT, HLEDAT, HODNOTA, HODNOTA.NA.TEXT, NAHRADIT, NAJÍT, STEJNÉ, ZLEVA, ZPRAVA
 - Z informačních: mohou se hodit všechny, ale žádná nebude nutná (vše půjde udělat nějak jinak).
 - Z matematických: ZAOKROUHLIT, USEKNOUT, NÁHČÍSLO, MOD.
- Existuje mnoho různých řešení, každé z nich využije nějaké z výše uvedených funkcí – čím víc jich budete znát, tím vyšší šance, že Vás něco napadne.

Jak pracovat s funkcemi?

- Zcela záměrně jsem Vám k funkcím na minulém slajdu nedal žádné informace. Co tedy s tím?
 - V menu „Vložit funkci“ vyhledejte odpovídající funkci, klikněte na její název a dole se Vám zobrazí popis.
 - Nedává-li Vám popis vůbec smysl, zkuste Google.
 - Vyzkoušejte, jestli funkce dělá to, co si myslíte, že dělá, případně si ujasněte nejasnosti v detailech.
 - Když už si myslíte, že všechno víte, zkuste si ještě několik testovacích hodnot.
- **Funkci skutečně chápete v případě, že pro každý vstup víte, co funkce vrátí.**
 - Výjimkou jsou funkce, které pracují s „náhodou“.

Cvičení

- Splníte-li následující cvičení s tím, že jim vážně rozumíte, pravděpodobně zvládnete písemku.
 - **Porozumění neznamená, že budete něco upravovat tak dlouho, dokud to náhodou nevyjde.**
- Napište funkci, která počítá zbytek po dělení hodnot ve dvou buňkách a ošetřuje veškeré chyby (tedy nejen dělení nulou). V případě chyby se vypíše chybová hláška, kterou si pro ten účel určíte.
- Napište funkci, která do buňky B1 napíše první čtyři znaky z buňky A1 a do buňky C1 napíše, jestli při „zkopírování“ A1 do B1 „zmizely“ nějaké znaky.
 - Vyzkoušejte více různých řešení.

Cvičení

- Napište funkci, která zjistí, jestli se vyskytuje slovo AHOJ v textu, který vznikne spojením textů v buňkách A1 a B1 (platí například, když je v buňce A1 slovo MESIAH a v buňce B1 slova OJ OD VOZU).
- Modifikujte předchozí příklad tak, aby funkce zjišťovala, zda je slovo AHOJ na pomezí buněk (tedy jestli začíná v jedné a končí v druhé).
- Napište funkci, která vrátí všechny znaky v buňce A1 za prvním výskytem znaku x (tedy například pokud je v A1 slovo ahojxneahoj, vrátí neahoj).
- **Ověřte si, zda chápete a rozumíte, co dělají všechny funkce uvedené v této prezentaci!!**

Rekapitulace

- Příští hodinu bude písemná práce.
 - Osobně jsem si zkoušel zadání cvičně napsat, trvalo mi to ani ne deset minut, Vy máte nejméně 90 minut.
 - V případě, že neovládáte základní věci z této i minulé hodiny, je možné to zadání nestihnout ani za několik hodin, protože si budete zbytečně přidělovat práci.
- K písemné práci můžete používat libovolné pomůcky včetně internetu (ne ke komunikaci), nemá tedy cenu se cokoliv učit z paměti (to už si můžete udělat raději rovnou tahák, který můžete používat).
- **Musíte nejen znát základy (například používání funkcí), ale také PŘEMÝŠLET!!**