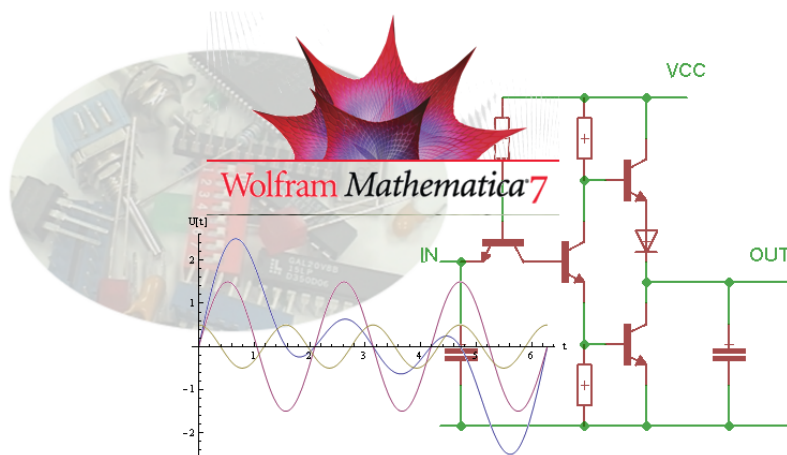




BI – CAO

vítejte na prvním cvičení předmětu
Číslicové a analogové obvody



Cvičící a kontakt: viz tabule nebo rozvrh na <https://courses.fit.cvut.cz/BI-CAO/teacher/index.html>

Vytvořeno na základě materiálů Miroslava Skrbka a Pavla Kubalíka.

Poslední úpravy: Martin Kohlík před ZS 2019/20

Co se naučíte v tomto předmětu ?

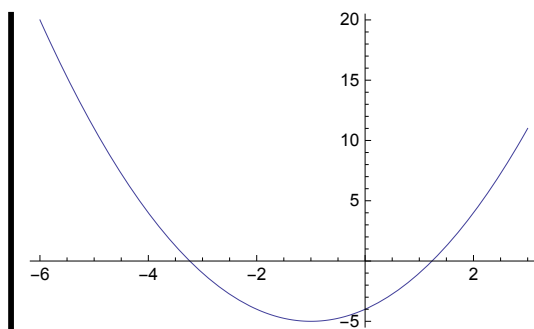
Naučíte se základům analogových a číslicových obvodů, návrh obvodů, jejich výpočty a simulaci.

Naučíte se pracovat v mocném programu Wolfram *Mathematica*, který budeme používat pro výpočty a simulaci obvodů

```
| Solve[x^2 + 2 x - 4 == 0, x] // N
```

```
| {{x -> -3.23607}, {x -> 1.23607}}
```

```
| Plot[x^2 + 2 x - 4, {x, -6, 3}]
```



◀ | ▶

Požadavky pro udělení zápočtu

Nutné podmínky

- absolvování tří testů
 - Test 1 (3. cvičení, max. 10 bodů)
 - Test 2 (7. cvičení, max. 20 bodů)
 - Test 3 (11. cvičení, max. 20 bodů)
- dosažení minimálně 25 bodů ze cvičení (testy + body navíc)

Body navíc

Max. 10 bodů navíc je možno získat za aktivitu ve cvičení. Na body za aktivitu nemá student nárok a o jejich udělení rozhoduje cvičící. Body za aktivitu může například dostat pět nejúspěšnějších řešitelů zadané úlohy.

Náhradní a opravné testy

- na náhradní test je nárok jen v odůvodněných případech (např. nemoc doložená potvrzením od lékaře) a vztahuje se na všechny 3 testy. O přijetí omluvy rozhodne cvičící.
- každý student má nárok na jeden opravný test (Test 1, Test 2, nebo Test 3). Opravný test nahrazuje (tedy ruší) výsledek řádného testu.
- Náhradní a opravné testy proběhnou ve zkouškovém období a bude vypsán jen jeden termín. Studenti se budou přihlašovat prostřednictvím informačního systému KOS na "Jednorázovou akci". CAO-T1(1. test), CAO-T2 (2. test) a/nebo CAO-T3 (3. test), aby deklarovali, který test budou opravovat/nahrazovat.

Wolfram Mathematica

Program Wolfram *Mathematica* je mocný nástroj pro symbolické počítání. Je ho možno použít jako

- kalkulačku
- maticový kalkulátor
- pro řešení lineárních rovnic s více proměnnými
- řešení diferenciálních rovnic
- symbolické derivování a integrování
- vykreslování grafů
- a spoustu dalších funkcí

◀ | ▶

Kde mohu pracovat s programem *Mathematica* ?

Ve cvičeních, kde je nainstalován na počítačích PC.

Program *Mathematica* si můžete (a my to velmi doporučujeme) nainstalovat na domácí počítač.

Instalační image (přípona .iso) si stáhnete na <http://download.cvut.cz> po přihlášení hlavním přístupovým heslem. Image vypálíte na CD nebo použijete utilitu isomount, který umí zpřístupnit image jako další disk bez nutnosti vypalování CD. Pak instalujte obvyklým způsobem.

Pro program potřebujete licenci. Pro získání licence se řiďte pokyny na stránce předmětu BI-CAO, použijte licenční servery leibniz.feld.cvut.cz.

Pokud budete mít s instalací problémy, sdělte to vašemu cvičímu a pokuste se problém vyřešit. Ideální je přinést přímo notebook s instalovaným programem (pokud máte).

PC v učebně NTK:PU1

Přihlášení os Linux

Pro přihlášení použijte přidělené celoškolské uživatelské jméno a heslo. Pokud jste se již v jiných cvičeních přihlašovali a změnili si heslo, tak musíte použít změněné heslo.

Program Mathematica spustíte tak, že vyhledáte v systémovém menu položku Console a v příkazovém řádku napíšete mathematica.

Soubory ukládáme na /home/stud/<vaše username>

Doporučujeme si vytvořit podadresář "cao", aby se vám soubory nepletly se soubory z jiných předmětů. Návod: v terminálu (Console) napíšeme: `cd ; mkdir cao`

Přihlášení os Windows

Pro přihlášení použijte přidělené celoškolské uživatelské jméno a heslo. Pokud jste se již v jiných cvičeních přihlašovali a změnili si heslo, tak musíte použít změněné heslo.

Program Mathematica najdete v nabídce Start.

Soubory ukládáme na disk H:/

Doporučujeme si vytvořit podadresář "cao", aby se vám soubory nepletly se soubory z jiných předmětů.

Mathematica - vytvoření notebooku, práce s buňkami

Notebook připomíná textový editor, který nám slouží jako rozhraní k výpočetnímu jádru programu *Mathematica*. Do notebooku píšeme matematické výrazy, komentáře v podobě formátovaného textu, zobrazují se nám zde výsledky výpočtů a grafy.

Notebook vytvoříme výběrem položky menu: **Menu→File→New→Notebook**

Notebook se skládá z buněk (Cell), které připomínají řádky (bloky) textu s proměnlivou výškou. Buňka může obsahovat více standardních řádků. Mezi buňkami přecházíme **kurzorovými šipkami nahoru/dolů**. Pozor! použití klávesy **Enter** znamená přidání řádku v rámci buňky, nikoliv přechod na další buňku.

Buňka je označena na pravé straně notebooku modrou skobičkou, na kterou můžeme kliknout myší a svinout buňku, rozvinout buňku, označit ji a pak klávesou delete vymazat.

Každá buňka má styl. Například: *input* - vstup pro výpočty, *out* - výstup výpočtu, *text* - text (dokumentace výpočtu), *title* - nadpis, ...

Buňky lze spojovat a rozdělovat.

Při práci s notebookem je vhodné si otevřít panel matematických vstupů. **Palettes → Ohter → Basic Math Input**

Vzhledem k tomu, že je program *Mathematica* napsan jako **Client ↔ Server**, jsou veškeré výpočty prováděny až po odeslání buňky na server s pomocí **SHIFT+ENTER**.

Čísla v programu *Mathematica*

Celá čísla (přesná)

$\{1, 2, 3, 4, -5, 1000\};$

Racionální čísla (přesná)

$\{\frac{4}{3}, -\frac{1}{3}, \frac{100}{330}\};$

Iracionální čísla (přesná)

$\{\sqrt{6}, e^5, \pi\};$

Komplexní (přesná, nepřesná pokud obsahují reálná čísla)

$\{1 + 3i, 6 - 2i, 8 - \frac{6}{5}i\};$

Reálná čísla (nepřesná)

$\{3.14, 1.41, 2.6545, -32.567\};$

Jednoduché výpočty

S programem *Mathematica* lze provádět i jednoduché výpočty.

$$3+4$$

$$5*8$$

$$2*\pi$$

Program *Mathematica* provádí všechny výpočty s ohledem na zachování přesnosti. Pokud chceme přesné číslo převést do nepřesné podoby (jako reálné číslo), použijeme funkci **N[2 π]**. Pokud chceme počítat "nepřesně" (např. pro zrychlení výpočtu) můžeme čísla zadávat jako reálná. Např.: místo 2* π zadáme 2.* π

Různé způsoby volání funkce

- Infixová forma 3 * 5
- Postfixová forma 2 π //N
- Prefixová forma N@ 2 π

Vstupy lze zadávat s pomocí panelu Basic Math Input, popřípadě je možné využít klávesovou zkratku

- Děleno $\frac{2}{3}$ lze zadat jako 2 CTRL + / a 3
- Mocnina 2³ lze zadat jako 2 CTRL + 6 a 3
- PI π lze zadat jako ESC p ESC
- Odmocnina $\sqrt{3}$ lze zadat jako CTRL + 2 a 3

Proměnné

Proměnné jsou například x, y, k, L , ale také $xx, x1$, kocka, pes23, ...

Proměnná začíná písmenem (snažte se používat malá písmena), za kterým následuje žádný, jeden nebo více znaků nebo číslic.

Proměnná nesmí začínat číslem, např. $2x$ má význam 2 krát x , tak jak je užíváno v matematice (oboru).

Nepoužívat podtržítka, jako je to zvykem v jazyce C, např. x_1 , podtržítko má v programu *Mathematica* zvláštní význam.

Pro jistotu **napoužívat** ani **indexy**, mohou vést k nepředpokládanému chování programu.

`{x, xx, kocka, pes, x1, xy3, a1, b787, abc4};`

Pozor ! xy neznamená x krát y , ale jméno proměnné xy . Násobení zapisujeme $x * y$. Jako násobení se také chápe $x y$ (mezera mezi x a y), to ale nepoužívejte, je to zdroj nepříjemných chyb.

Jména proměnných volte rozumně, spíše více písmen dávajících smysl (rovnice1, vstup, ...), zvyšuje to čitelnost programu.

Operátor =

Pro pochopení operátoru si představte, že program *Mathematica* obsahuje tabulku jmen proměnných a jejich hodnot (tabulku symbolů). Každá položka (řádek) obsahuje jméno proměnné a její hodnotu.

x = 1; (* do proměnné x přiřad' jedničku *)

Např. po provedení x=1 obsahuje tabulka řádek pro x (na začátku předpokládáme prázdnou tabulku symbolů).

proměnná	hodnota
x	1

y = 2; (* do proměnné y přiřad' dvojku *)

proměnná	hodnota
x	1
y	2

Nyní jsou v tabulce symbolů dvě proměnné x a y s přazenými hodnotami.

◀ | ▶

Změna obsahu proměnné

x = 1;

y = 2;

Tabulka symbolů po provedení výše uvedených přiřazení

proměnná	hodnota
x	1
y	2

x = 5;

Nyní jsme změnilí obsah proměnné x.

proměnná	hodnota
x	5
y	2

◀ | ▶

Vyhodnocení jednoduchých výrazů

x = 1;

y = 2;

Tabulka symbolů po provedení výše uvedených přiřazení

proměnná	hodnota
x	1
y	2

z = x + y;

Výraz $x+y$ na pravé straně operátoru přiřazení se vyhodnotí tak, že se *Mathematica* pro x podívá do tabulky symbolů a nahradí ho jeho hodnotou (1 v našem případě), totéž udělá pro y a pak teprve provede součet. Pak bude tabulka symbolů vypadat takto:

proměnná	hodnota
x	1
y	2
z	3

◀ | ▶

Funkční program

```
x = 1;  
y = 2;  
z = x + y;  
z (* vytiskni obsah z *)  
3
```

Zadání k vlastnímu řešení:

Vypočtěte hodnotu výrazu $x^2 + 5y - 6z$ pro hodnoty $x=10$, $y=25$, $z=8$
Výsledek bude uložen v proměnné r .

◀ | ▶

Záludnosti při vyhodnocení buněk v notebooku

Buňky v notebooku se vyhodnocují na vyžádání, a to buď

- jednotlivě (v každé buňce stiskneme klávesy Shift-Enter)

pořadí vyhodnocení je dáno pořadím výběru buněk k vyhodnocení. Pokud vyhodnocujeme buňky na přeskáčku, může se stát, že dostaneme nesmyslné výsledky, protože si jádro vytváří tabulku symbolů v pořadí tak, jak vyhodnocujeme buňky a ne jak jsou zapsány v notebooku. Navíc je možné některé pravidlo vyhodnotit vícekrát. Pokud děláme navíc v notebooku zásadnější změny, tak i přes to, že veškeré změny odesíláme do jádra (Shift-Enter), tak stav tabulky symbolů před změnami nám může způsobit nesmyslný výsledek.

Provedte:

Menu→Evaluation→QuitKernel→Local (zajistí vynulování tabulky symbolů, začínáme tedy s "čistým stolem")

Shift+Enter na buňce $z=x+y$

výsledek je $x+y$ protože ani x ani y nebylo zatím definováno ($x=1$ a $y=2$ nebylo ještě posláno do jádra).

Shift+Enter na buňce $y=2$; a potom na $z = x+y$

výsledek je $x+2$ protože y má hodnotu 2

Shift+Enter na buňce $x=1$; a potom na $z = x+y$

výsledek je 3, který jsme očekávali

- hromadně (**Menu→Evaluation→EvaluateNotebook**)

Vyhodnotí všechny buňky v notebooku směrem od začátku do konce. Toto je významné, ale stále jen částečné řešení předchozího problému. Stále ještě hrozí, že před vyvoláním **Menu→Evaluation→EvaluateNotebook** nebyla tabulka symbolů prázdná a zvláště pokud používáme jednu proměnnou (jedno jméno) vícenásobně pro různé a nesouvisející výrazy, můžeme se stejně dobrat chybných výsledků. **Pokud chceme mít jistotu, pak musíme inicializovat kernel (Menu→Evaluation→QuitKernel→Local) a vyhodnotit všechny buňky v notebooku Menu→Evaluation→EvaluateNotebook.**

Pozor! Pokud máme otevřeno více notebooků, pak všechny využívají jeden kernel. Tzn. proměnná nadefinovaná v jednom notebooku je viditelná v jiném notebooku.

Není to však pravidlem - v některých operačních systémech se při otevření uloženého notebooku spustí i nový samostatný kernel.

Řešení problémů s vyhodnocováním buněk

Před každou skupinou buněk, která řeší nějaký matematický problém vymažeme funkci `ClearAll["Global`*"]`; hodnoty všech proměnných v tabulce symbolů.

```
ClearAll["Global`*"];
x = 1;
y = 2;
z = x + y
3
```

Pak pokud použijeme funkci **Menu→Evaluation→Evaluate Notebook** máme jistotu, že se nám všechny buňky vyhodnotí ve správném pořadí a navíc budou správně inicializovány.

Toto si zvykneme psát do našich notebooků, abychom stále nenaráželi na problémy s vyhodnocováním buněk.

V případě, že je v notebooku více různých menších příkladů (typicky testy), umístíme každý z nich do jedné samostatné buňky a na její začátek vložíme funkci `ClearAll["Global`*"]`;

`ClearAll` můžeme použít i v případě, že potřebujeme smazat hodnoty pouze u vybraných proměnných:

```
ClearAll["Global`*"];
x = 1;
y = 1;
z = x + y;
ClearAll[x, y];
x
y
z (*z jsme nevymazali, tudiz si svou hodnotu pamatuje*)
x
y
2
```

Funkce (vestavěné)

Funkce v programu *Mathematica* hrají stejnou roli jako funkce v matematice.

Jsou zde pouze odlišnosti v zápisu. Např. $\sin(x)$ se zapíše jako `Sin[x]`.

Vestavěné funkce začínají vždy velkým písmenem a argumenty (parametry) jsou uzavřeny v hranatých závorkách.

Funkce má určitý počet pevně určených parametrů a ostatní jsou volitelné a nezávisí na jejich pořadí. Volitelné parametry se zapisují jako pravidla, např. `PlotRange->All`.

Např. funkce `Sin[x]` má jeden pevný parametr. Funkce `Plot[funkce, rozsah, PlotRange->All, ...]` má dva pevné parametry funkce a rozsah, další parametry jsou volitelné.

Význam jednotlivých parametrů funkcí nalezneme v helpu.

? Sin

`Sin[z]` gives the sine of z . >>

? Plot

`Plot[f, {x, x_{min} , x_{max} }]` generates a plot of f as a function of x from x_{min} to x_{max} .

`Plot[{ f_1 , f_2 , ...}, {x, x_{min} , x_{max} }]` plots several functions f_i . >>

Pokud neznáme plně jméno funkce stačí naznačit (napsat jen část) např. `Arc` a pak Ctrl-k. *Mathematica* napoví (objeví se pop-up menu s možnostmi).

V aktuálních verzích *Mathematicy* se toto menu ukazuje během psaní názvu funkce už automaticky.

Funkce definované uživatelem

Uživatel programu *Mathematica* si může definovat vlastní funkci.

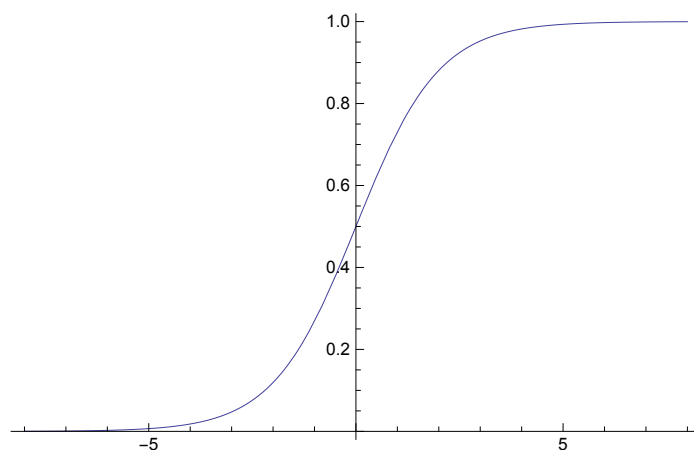
Parametry na levé straně přiřazení musí končit podtržítkem `x_`, `yy_`, případně `x:_` nebo `yy:_`. Jedná se tzv. formální parametry, za které se v místě užití dosadí parametry skutečné. Formální parametry na pravé straně přiřazení podtržítka nemají.

Užívá se zde jiného operátoru přiřazení (`:=`). Ten je interpretován odlišně, oproti operátoru `=`. U operátoru `=` je k levé straně (jméno proměnné, funkce, ...) v tabulce symbolů přiřazena hodnota (vyhodnocený výraz na pravé straně) okamžitě. U operátoru `:=` dojde k nahrazení levé strany pravou až v době užití, kdy je znám ke každému formálnímu parametru odpovídající parametr skutečný. Např. `sigmolda[a+3, 1]` se nahradí $1 / (1 + e^{-1 \cdot (a+3)})$, kde výraz `a+3` a konstanta `1` jsou skutečnými parametry.

Definujme logistickou funkci zvanou Sigmoida.

```
sigmolda[x_, y_] := 1 / (1 + e-x*y);
```

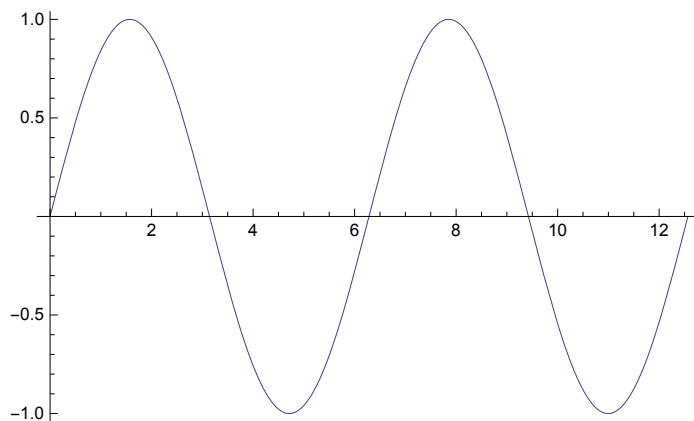
```
Plot[sigmolda[a, 1], {a, -8, 8}]
```



Kreslení grafů

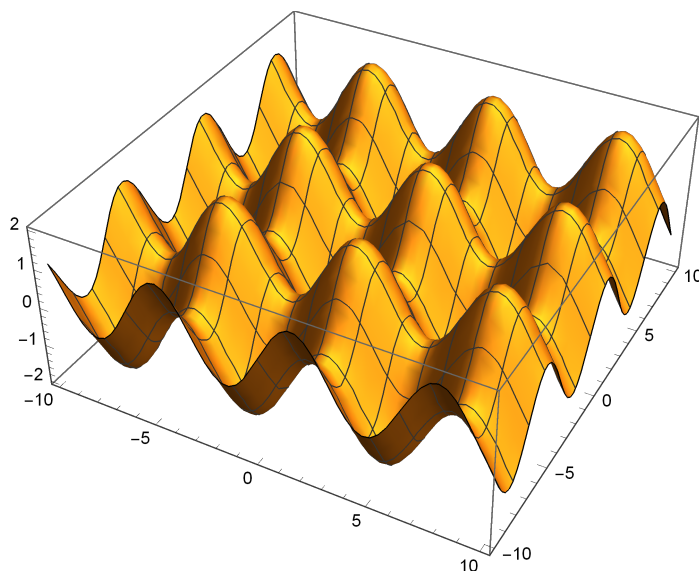
Pro kreslení grafů (2D) používáme funkci `Plot`. První parametr je funkce, kterou chceme vykreslit. Druhý parametr je obor hodnot funkce pro zvolenou nezávislou proměnnou. Druhý parametr je vždy seznam obsahující po řadě proměnnou, minimální hodnotu a maximální hodnotu.

`Plot[Sin[x], {x, 0, 4 π}]`



Pro kreslení grafů (3D) používáme funkci `Plot3D`. První parametr je funkce, kterou chceme vykreslit. Druhý a třetí parametr jsou obory hodnot funkcí pro zvolené nezávislé proměnné. Druhý i třetí parametr jsou vždy seznamy obsahující po řadě proměnnou, minimální hodnotu a maximální hodnotu.

`Plot3D[Sin[x] + Sin[y], {x, -10, 10}, {y, -10, 10}]`



Existují i další typy grafů (např. `LogPlot`, `ParametricPlot`, `ListPlot`, `ListPlot3D`, apod.) Detaily o těchto typech grafů lze nalézt v nápovědě.