

# Cvičení 1

## Modelování systémů a procesů

Mgr. Lucie Kárná, PhD

karna@fd.cvut.cz

18. února 2020

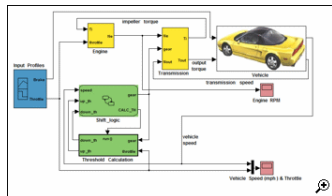
# 1 Organizace cvičení

## 2 Matlab

- Začínáme
- Základní operace
- Základní funkce

## 3 Simulink

- Princip práce v Simulinku
- Jednoduché modely v Simulinku
- Souhrn



# Webová stránka předmětu

<http://zolotarev.fd.cvut.cz/msp/>

- program cvičení a přednášek
- termíny testů
- pravidla
- studijní materiály, videa
- zadání domácích úloh
- odevzdání domácích úloh
- průběžné hodnocení

# Zápočet

Nezávazný výtah z webu

Nejméně 25 bodů ze 40 možných

- 4 body za tři automatické domácí úkoly
- 10 bodů za 3 malé průběžné testy
- 12 bodů za dva praktické testy (Matlab, Simulink)
- 14 bodů za velký početní test (minimálně 7)

plus možné bonusové body za aktivitu na cvičeních

25 bodů ⇒ zkouška za **E**

30 bodů ⇒ zkouška za **D**

35 bodů ⇒ možnost domácího zadání místo zkoušky

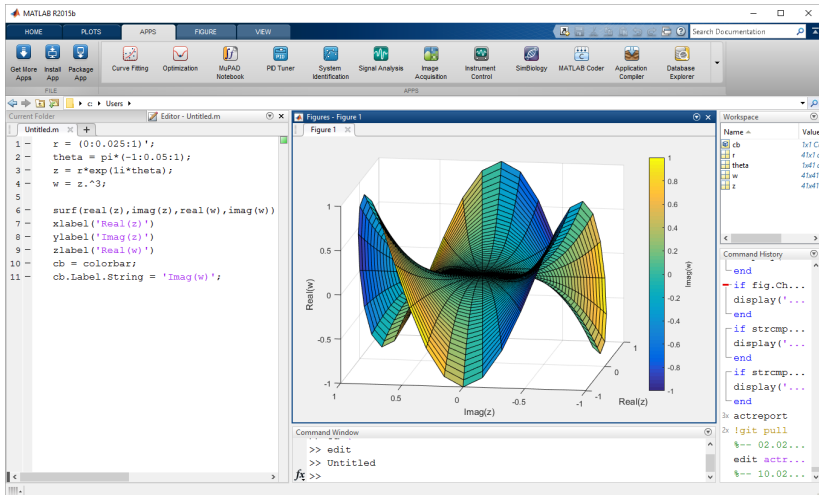
Opravy: jen velký početní test + jeden průběžný test

Omluvy: **nezneužívat**, jinak pouze s potvrzením (dokladem)

# Předpokládané vstupní znalosti

- středoškolská matematika
  - úpravy složených výrazů
  - kvadratické rovnice
  - komplexní čísla,
- algebra
  - maticový počet
  - vlastní čísla
  - soustavy lineárních rovnic
  - parciální zlomky
- kalkulus
  - číselné řady
  - integrace a derivace základních funkcí
- základy programování a algoritmizace

## Matlab



# Matlab

## instalace Matlabu

stáhnout z <https://download.cvut.cz/>  
pouze z IP adres domény ČVUT

## Matlab mluví anglicky

- přepnout klávesnici na ENG/US
- desetinná *tečka*
- **nepoužívat** háčky, čárky, mezery, speciální znaky atd.  
v názvech proměnných ani souborů

# Pohodlí při práci

## UNIX-like prostředí

- rozlišujeme malá a VELKÁ písmena
- doplnění slova: tabulátor
- zkopírování minulého příkazu: šipka nahoru
- ukončení probíhajícího výpočtu:  $\wedge C$



# Pohodlí při práci

## UNIX-like prostředí

- rozlišujeme malá a VELKÁ písmena
- doplnění slova: tabulátor
- zkopírování minulého příkazu: šipka nahoru
- ukončení probíhajícího výpočtu:  $\wedge C$

- okno Workspace - přehled proměnných
- nastavit pracovní adresář
- **červená odpověď** = chyba

# Základní operace

## Matlab jako kalkulačka

```
>> 1320 / 63
```

```
% za znakem '%' je komentář
```

# Základní operace

## Matlab jako kalkulačka

```
>> 1320 / 63                                % za znakem '%' je komentář  
ans = 20.9524                               % proměnná 'ans' = odpověď  
>> p = ans - 20                             % proměnnou 'ans' lze dále využívat
```

# Základní operace

## Matlab jako kalkulačka

```
>> 1320 / 63                % za znakem '%' je komentář  
ans = 20.9524              % proměnná 'ans' = odpověď  
>> p = ans - 20           % proměnnou 'ans' lze dále využívat  
>> a = 1 + 1;             % potlačení výstupu na obrazovku
```

# Základní operace

## Matlab jako kalkulačka

```
>> 1320 / 63                                % za znakem '%' je komentář
ans = 20.9524                                % proměnná 'ans' = odpověď
>> p = ans - 20    % proměnnou 'ans' lze dále využívat
>> a = 1 + 1;      % potlačení výstupu na obrazovku
>> a = a + 1
>> a = 3
```

# Vektory 1

## Zadávání vektorů

```
>> u = [1 2 3 4 5]
```

```
% vycet prvku
```

# Vektory 1

## Zadávání vektorů

```
>> u = [1 2 3 4 5]
```

```
>> x = 1:5
```

```
% vycet prvku
```

```
% notace s dvojteckou
```

# Vektory 1

## Zadávání vektorů

```
>> u = [1 2 3 4 5]                                % vycet prvku  
>> x = 1:5                                         % notace s dvojteckou  
>> y = 0:pi/4:pi
```



# Vektory 1

## Zadávání vektorů

```
>> u = [1 2 3 4 5] % vycet prvku  
>> x = 1:5 % notace s dvojteckou  
>> y = 0:pi/4:pi
```

## Čtení a zapisování prvků vektoru

```
>> u = [1 3 5 7];
```

# Vektory 1

## Zadávání vektorů

```
>> u = [1 2 3 4 5] % vycet prvku  
>> x = 1:5 % notace s dvojteckou  
>> y = 0:pi/4:pi
```

## Čtení a zapisování prvků vektoru

```
>> u = [1 3 5 7];  
>> x = u(2)
```

# Vektory 1

## Zadávání vektorů

```
>> u = [1 2 3 4 5] % vycet prvku
>> x = 1:5 % notace s dvojteckou
>> y = 0:pi/4:pi
```

## Čtení a zapisování prvků vektoru

```
>> u = [1 3 5 7];
>> x = u(2)
>> u(4) = 9;
```

# Vektory 2

## Řádkový a sloupcový vektor

```
>> x = [0.0:0.1:0.5]'           % apostrof = transpozice
```

# Vektory 2

## Řádkový a sloupcový vektor

```
>> x = [0.0:0.1:0.5] '           % apostrof = transpozice  
>> y = exp(-x).*cos(x);        % člen po členu - s tečkou
```

# Vektory 2

## Řádkový a sloupcový vektor

```
>> x = [0.0:0.1:0.5]'           % apostrof = transpozice
>> y = exp(-x).*cos(x);       % člen po členu - s tečkou
>> [x y]                       % matice (= tabulka)
```

# Vektory 2

## Řádkový a sloupcový vektor

```
>> x = [0.0:0.1:0.5]'           % apostrof = transpozice
>> y = exp(-x).*cos(x);        % člen po členu - s tečkou
>> [x y]                        % matice (= tabulka)
```

## Skalární součin

```
>> u = [2 -3 1];
>> v = [-3 1 2];
>> u*v      % chyba - matice 1x3 krat 1x3 nelze nasobit
```

# Vektory 2

## Řádkový a sloupcový vektor

```
>> x = [0.0:0.1:0.5]'           % apostrof = transpozice
>> y = exp(-x).*cos(x);        % člen po členu - s tečkou
>> [x y]                         % matice (= tabulka)
```

## Skalární součin

```
>> u = [2 -3 1];
>> v = [-3 1 2];
>> u*v      % chyba - matice 1x3 krat 1x3 nelze nasobit
>> w = x*v'  % skalární součin - 1x3 krat 3x1
```



# Matice

- indexování řádků a sloupců od jedničky

# Matice

- indexování řádků a sloupců od jedničky
- zadávání matice výčtem prvků:

```
>> A = [1 2; 3 4; 5 6] % řádky odděluje středník
```

# Matice

- indexování řádků a sloupců od jedničky
- zadávání matice výčtem prvků:

```
>> A = [1 2; 3 4; 5 6] % řádky odděluje středník
```

```
>> A(2,1) % prvek  $A_{21} = 3$ ; indexy odděluje čárka!
```

# Matice

- indexování řádků a sloupců od jedničky
- zadávání matice výčtem prvků:

```
>> A = [1 2; 3 4; 5 6] % řádky odděluje středník  
>> A(2,1) % prvek  $A_{21} = 3$ ; indexy odděluje čárka!  
>> A(:,1) % první sloupec
```

# Matice

- indexování řádků a sloupců od jedničky
- zadávání matice výčtem prvků:

```
>> A = [1 2; 3 4; 5 6] % řádky odděluje středník  
>> A(2,1) % prvek  $A_{21} = 3$ ; indexy odděluje čárka!  
>> A(:,1) % první sloupec  
>> A(2,:) = [] % vymaže 2. řádek
```

# Matrice

- indexování řádků a sloupců od jedničky
- zadávání matice výčtem prvků:

```
>> A = [1 2; 3 4; 5 6] % řádky odděluje středník  
>> A(2,1) % prvek  $A_{21} = 3$ ; indexy odděluje čárka!  
>> A(:,1) % první sloupec  
>> A(2,:) = [] % vymaže 2. řádek
```

- násobení matic:

```
>> A = [1 2; -3 1]  
>> B = [3 -1; -2 3]  
>> A*B
```

# Matice

- indexování řádků a sloupců od jedničky
- zadávání matice výčtem prvků:

```
>> A = [1 2; 3 4; 5 6] % řádky odděluje středník  
>> A(2,1) % prvek  $A_{21} = 3$ ; indexy odděluje čárka!  
>> A(:,1) % první sloupec  
>> A(2,:) = [] % vymaže 2. řádek
```

- násobení matic:

```
>> A = [1 2; -3 1]  
>> B = [3 -1; -2 3]  
>> A*B
```

- násobení po prvcích – tečková konvence:

```
>> A.*B
```

# Další dovednosti

## Domácí úkol

Nastudovat *Jemný úvod do Matlabu a Simulinku*

na stránkách předmětu, sekce *Cvičení*, materiály pro 1. cvičení



# Základní funkce

## Obecné funkce

`help` on-line nápověda

`who` seznam proměnných

`clear` zruší všechny proměnné

`clc` vymaže obrazovku

## Matematické funkce

`exp` exponenciální funkce –  $e^x$

`x^a` obecná mocnina –  $x^a$

`sqrt` druhá odmocnina (**s**quare **r**oot) –  $\sqrt{x}$

## Vektorové funkce

`length` délka vektoru

`roots` výpočet kořenů polynomu

## Vektorové funkce

`length` délka vektoru

`roots` výpočet kořenů polynomu

## Příklad – kořeny polynomu

- Zadání: najděte kořeny polynomu  $p(x) = 3x^3 + 2x + 1$

## Vektorové funkce

`length` délka vektoru

`roots` výpočet kořenů polynomu

## Příklad – kořeny polynomu

- Zadání: najděte kořeny polynomu  $p(x) = 3x^3 + 2x + 1$
- Řešení:

```
>> p = [3, 0, 2, 1]    % vektor = koeficienty polynomu  
>> roots(p)           % vrati koreny
```

## Maticové funkce

`size` dimenze matice

`zeros(m,n)` nulová matice (m,n)

`ones(m,n)` matice (m,n) jedniček

## Maticové funkce

`size` dimenze matice

`zeros(m,n)` nulová matice (m,n)

`ones(m,n)` matice (m,n) jedniček

`eye(m)` jednotková matice (m,m)

`rand(m,n)` matice (m,n) náhodných čísel

## Maticové funkce

`size` dimenze matice

`zeros(m,n)` nulová matice (m,n)

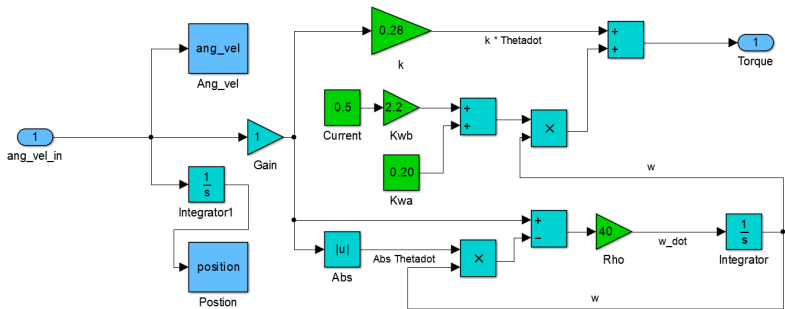
`ones(m,n)` matice (m,n) jedniček

`eye(m)` jednotková matice (m,m)

`rand(m,n)` matice (m,n) náhodných čísel

`eig` výpočet vlastních čísel matice

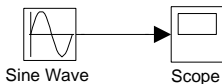
## Simulink





# Příklad: zobrazení sinusové vlny

## Model

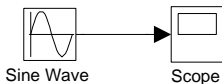


## Použité bloky

- Sources → Sine Wave
- Sinks → Scope

# Příklad: zobrazení sinusové vlny

## Model



## Použité bloky

- Sources → Sine Wave
- Sinks → Scope

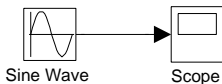
## Blok Sine Wave

parametry:

- frekvence
- fáze
- amplituda

# Příklad: zobrazení sinusové vlny

## Model



## Použité bloky

- Sources → Sine Wave
- Sinks → Scope

## Parametry simulace

- Start Time
- Stop Time
- Solver Type (Variable/Fixed Step)
- Step Time (auto/hodnota)

## Blok Sine Wave

parametry:

- frekvence
- fáze
- amplituda

# Kružnice

## Rovnice

$$x = r \sin t,$$

$$y = r \cos t,$$

$$t \in \langle 0, 2\pi \rangle,$$

$$r > 0.$$

# Kružnice

## Rovnice

$$x = r \sin t,$$

$$y = r \cos t,$$

$$t \in \langle 0, 2\pi \rangle,$$

$$r > 0.$$

## Sources → Clock

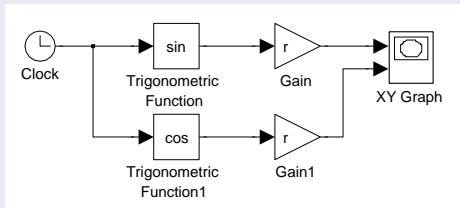


Clock

parametry:

nenastavujeme

## Model



## Sinks → XY Graph

parametry:

- Xmin, Xmax: rozsah na ose X
- Ymin, Ymax: rozsah na ose Y

# Kružnice

## Blok Math Operations → Gain



- parametry: hodnota činitele
- zde hodnotu  $r$  zadáme v Matlabu:  
`>> r=0.6`

## Blok Math Operations → Trigonometric Function

- sinus, cosinus, tangens, ...
- hyperbolický sinus, cosinus, ...
- ...

# Archimédova spirála

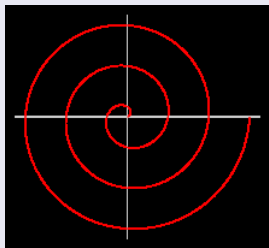
## Rovnice

$$x = t \sin t,$$

$$y = t \cos t,$$

$$t \in \langle 0, \infty \rangle.$$

## Model a výsledek



# Archimédova spirála

## Rovnice

$$x = t \sin t,$$

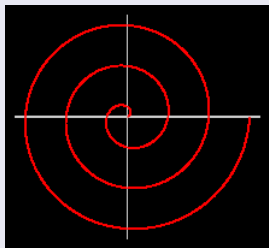
$$y = t \cos t,$$

$$t \in \langle 0, \infty \rangle.$$

Blok Math Operations → Product

parametry: počet vstupů

## Model a výsledek





# Archimédova spirála

## Rovnice

$$x = t \sin t,$$

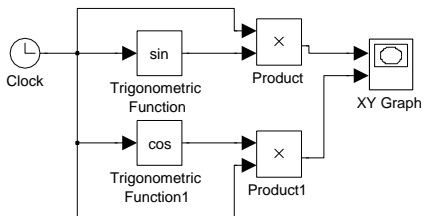
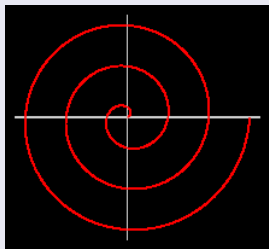
$$y = t \cos t,$$

$$t \in \langle 0, \infty \rangle.$$

Blok Math Operations → Product

parametry: počet vstupů

## Model a výsledek



# Logaritmická spirála

## Rovnice

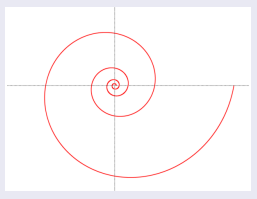
$$x = e^{-kt} \sin t,$$

$$y = e^{-kt} \cos t,$$

$$t \in \langle 0, \infty \rangle,$$

$$k > 0.$$

## Výsledek



# Logaritmická spirála

## Rovnice

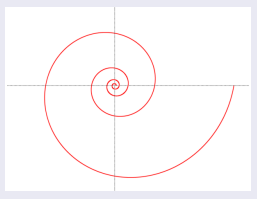
$$x = e^{-kt} \sin t,$$

$$y = e^{-kt} \cos t,$$

$$t \in \langle 0, \infty \rangle,$$

$$k > 0.$$

## Výsledek



## Blok Math Operations → Math Function

**exp** exponenciální funkce  $e^u$

**log** přirozený logaritmus  $\ln u$

**reciprocal** převrácená hodnota  $1/u$

**pow** obecná mocina  $u^v$

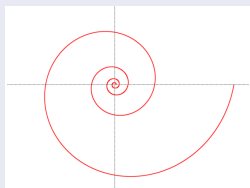
...

# Logaritmická spirála

## Rovnice

$$\begin{aligned}x &= e^{-kt} \sin t, \\y &= e^{-kt} \cos t, \\t &\in \langle 0, \infty \rangle, \\k &> 0.\end{aligned}$$

## Výsledek



## Blok Math Operations → Math Function

`exp` exponenciální funkce  $e^u$

`log` přirozený logaritmus  $\ln u$

`reciprocal` převrácená hodnota  $1/u$

`pow` obecná mocina  $u^v$

...

## Nastavení

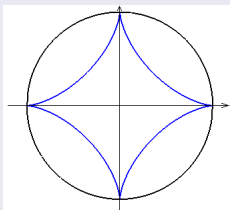
- v Matlabu položíme `>> k=0.05`
- konfigurace simulace: pevný krok 0.01.

# Asteroida

## Rovnice

$$\begin{aligned}x &= \sin^3 t, \\y &= \cos^3 t, \\t &\in \langle 0, 2\pi \rangle.\end{aligned}$$

## Výsledek

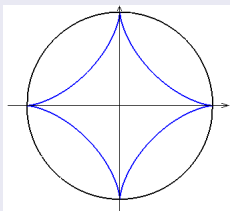


# Asteroida

## Rovnice

$$\begin{aligned}x &= \sin^3 t, \\y &= \cos^3 t, \\t &\in \langle 0, 2\pi \rangle.\end{aligned}$$

## Výsledek



## Blok

Math Operations

→ Math Function

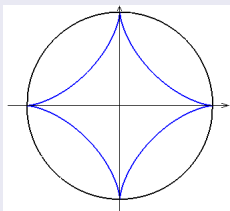
`pow` obecná mocina  $u^v$

# Asteroida

## Rovnice

$$\begin{aligned}x &= \sin^3 t, \\y &= \cos^3 t, \\t &\in \langle 0, 2\pi \rangle.\end{aligned}$$

## Výsledek



## Blok

Math Operations

→ Math Function

pow obecná mocina  $u^v$

Blok Sources → Constant

- nastavíme 3

# Nově probrané Simulinkové bloky

## Sources

- Sine Wave
- Clock
- Constant

## Sinks

- Scope
- XY Graph

## Math Operations

- Trigonometric Function
- Gain
- Product
- Math Function

## Signal Routing

- Mux



THE END

