



**FAKULTA
STAVEBNÍ
ČVUT V PRAZE**

102FY_2
Fyzika 2 G



Ing. Jan Trejbal, Ph.D.

Katedra fyziky
FSv ČVUT

Jan.trejbal@fsv.cvut.cz

<http://people.fsv.cvut.cz/~trejbj4/index.html>

Opakování: operace s vektory a maticemi

% Sloupcovy vektor

a = [2; 5; 7; 10];

b = [4

3

-5

11]

% Radkový vektor

c = [1 2 4 8 10];

d = [-12, 12, 5, -5]

% Skalar

e = 5;

% Matice forma zapsu 1

A = [1 8 9

4 2 -5

0 2 -9];

% Matice forma zapsu 2

B = [7 8 9; 1 0 -5; 2 7 -9];

% Matice forma zapsu 3

C = [10, 12, -15; 7, 0, -3; 12, -11, 0];

Opakování: dimenze vektoru a matice

```
A = [1 8 9
4 2 -5
0 2 -9];
```

```
B = [7 8 9; 1 0 -5; 2 7 -9];
```

```
C = [10, 12, -15; 7, 0, -3; 12, -11, 0];
```

```
a = [2; 5; 7; 10];
```

```
b = [4
3
-5
11];
```

```
c = [1 2 4 8 10];
```

```
d = [-12, 12, 5, -5];
```

```
e = 5;
```

```
%%
```

```
%Dimenze prvku uvedenych vyse:
```

```
D = size (A)
```

```
E = size (B)
```

```
F = size (C)
```

```
f = size (a)
```

```
g = size (b)
```

```
h = size (c)
```

```
i = size (d)
```

```
j = size (e)
```

Opakování: délka vektoru

```
a = [2; 5; 7; 10];
```

```
b = [4
```

```
3
```

```
-5
```

```
11];
```

```
c = [1 2 4 8 10];
```

```
d = [-12, 12, 5, -5];
```

```
e = 5;
```

```
%%
```

```
% Delka vektoru uvedenych vyse:
```

```
f = length (a)
```

```
g = length (b)
```

```
h = length (c)
```

```
i = length (d)
```

```
j = length (e)
```

Opakování: textové řetězce

% Ukazka textu - funkce disp

Znacka_automobilu = 'Ford ';

Model = 'Mondeo ';

Motorizace = '2.2 TDCi ';

Vykon = '129 kW';

% Vypis retezce na terminal

disp (Znacka_automobilu)

disp (['Prodané vozidlo:', Znacka_automobilu, Model])

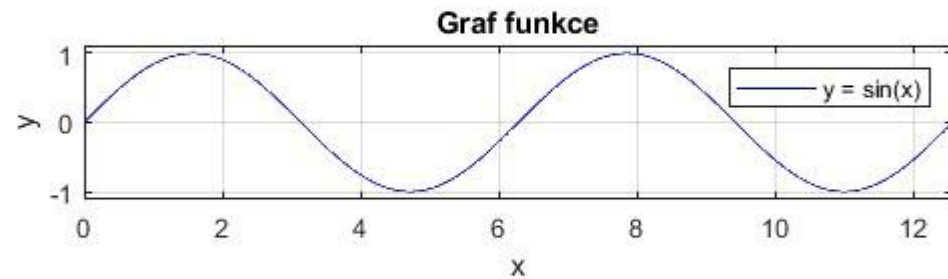
Matematické a statistické funkce

abs	% absolutni hodnota
sin, sind	% sinus (radiany), sinus (stupne)
cos, cosd	% cosinus
log, log10	% prirodzeny/dekadicky logaritmus
round	% zaokrouhleni na nejbližsi cele
fix	% zaokrouhleni na nizsi cele
sort	% razeni matice
sum	% soucet prvku matice
min, max	% minimum, maximum
mean, median	% prumer, median
std, var	% smerodatna odchylka, variance

- 2D
- Každý graf musí povinně obsahovat:
 - Název
 - Popis osy x
 - Popis osy y
 - Jednotky osy x
 - Jednotky osy y
 - Legendu
 - (Někdy je vhodné přidat předpis funkce)
- Připomenutí
 - linspace (od, do, počet prvků)
 - close all

```
x = linspace(0,4*pi,1000);
y = sin(x);
```

```
figure(1)           % okno grafu
plot(x,y,'b')       % modra linie
xlabel('x')          % popis osy x
ylabel('y')          % popis osy y
legend('y = sin(x)') % legenda
title('Graf funkce') % nadpis
grid on             % mřížka
axis equal           % rovnomerne osy
xlim([0 4*pi])      % limity osy x
ylim([-1.1 1.1])    % limity osy y
```



- Připravte graf funkce okamžité výchylky HB při harmonickém kmitání dle následujícího zadání:
 - Amplituda $A = 20 \text{ cm}$
 - Frekvence $f = 2 \text{ s}^{-1}$
 - Fázový posun $\varphi = \pi/3$

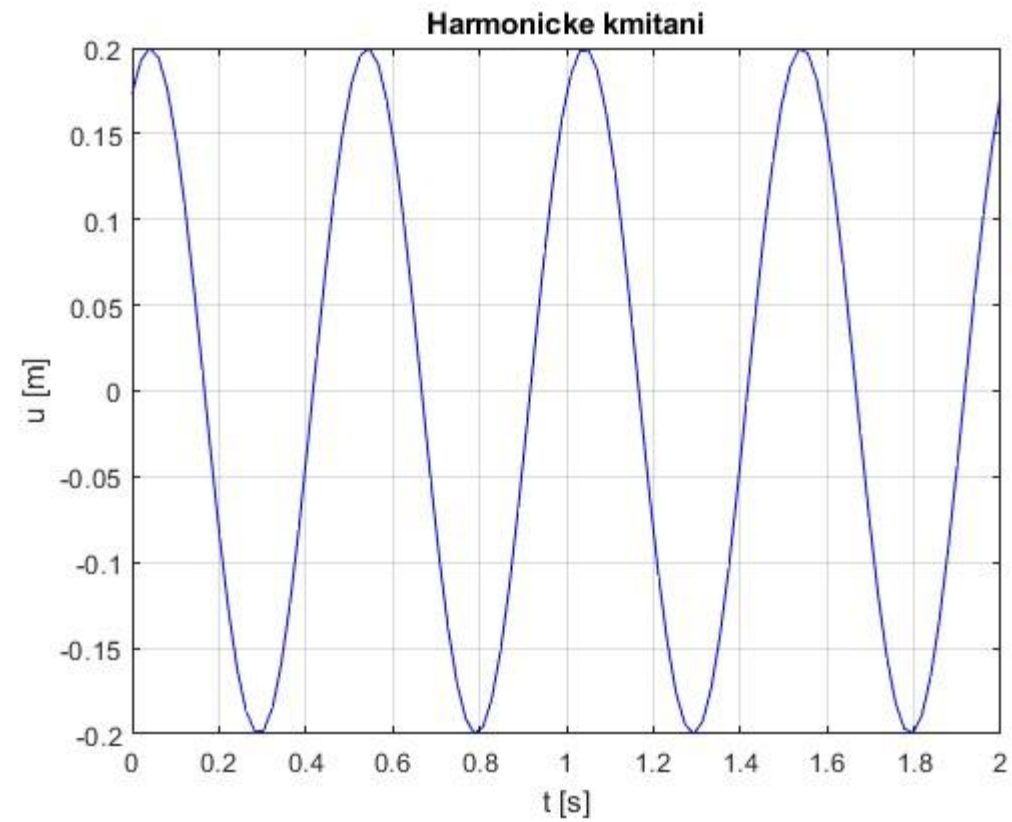
- Řešení grafu harmonického kmitání

```

A = 0.2;           % Amplituda
Phi = pi/3;       % Fazovy posun
f = 2;            % Frekvence
Omega = 2*pi*f;

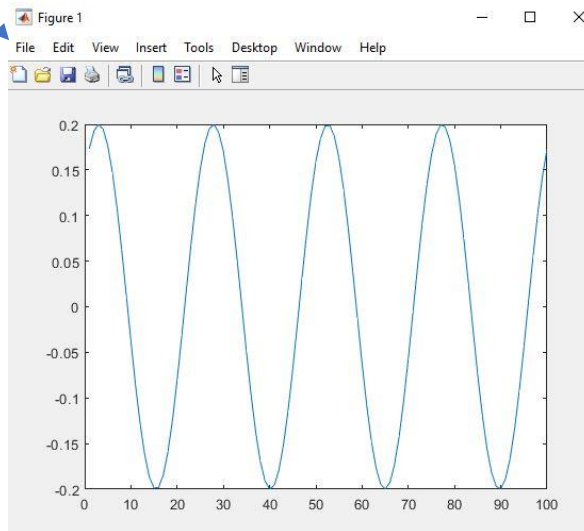
t = linspace(0,2,100); % Rozdeleni nezávisle funkce o 0 do 10 s, celkem 1000 hodnot
u_t = A*sin(Omega*t + Phi); % Zavisla funkce
figure(1)         % okno grafu
plot(t,u_t,'b-')  % modra linie
xlabel('t [s]')   % popis osy x
ylabel('u [m]')  % popis osy y
title('Harmonicke kmitani') % nadpis
grid on          % mřížka
%axis equal      % rovnomerne osy - vypnuto
xlim([0 2])     % limity osy x
ylim([-0.2 0.2]) % limity osy y

```



Grafy

- Další možnosti tvorby grafu – z Workspace
 - Označit proměnnou ve Workspace
 - Kliknou na záložku Plots
 - V okně Figure lze graf editovat
 - Z grafu lze zpětně generovat kód



Workspace

Name	Value
A	0.2000
f	2
Omega	12.5664
Phi	1.0472
t	1x100 double
u_t	1x100 double

```

1  clc
2  clear variables
3  close all
4
5  A = 0.2; % Amplituda
6  Phi = pi/3; % Fazovy posun
7  f = 2; % Frekvence
8  Omega = 2*pi*f;
9
10 t = linspace(0,2,100); % Rozdeleni nezavisle funkce o 0 do 10 s, celkem 1000 hodnot
11 u_t = A*sin(Omega*t + Phi); % Zavisla funkce
12 figure(1) % okno grafu
13 plot(t,u_t,'b-') % modra linie
14 xlabel('t [s]') % popis osy x
15 ylabel('u [m]') % popis osy y
16 title('Harmonicke kmitani') % nadpis
17 grid on % mrazka
18 %axis equal % rovnomerne osy - vypnuto
19

```

Command Window

```

>> plot(u_t)
fx >>

```

Grafy - další typy

- Liniové
 - plot, ploty, semilogx, semilogy, loglog, area, errorbar, plot 3, comet
- Scatter
 - scatter, scatter3
- Koláčové grafy
 - pie, pie3
- Histogramy
 - hist, rose
- Prostorové grafy
 - surf, surrfc, surf1, mesh, meshc, meshz, waterfall, ribbon, contour3



Načítání dat z externích souborů

- Načtení dat z textového (*.txt) souboru pomocí funkce fid a fopen
- Zdrojový soubor musí být umístěn v pracovním adresáři

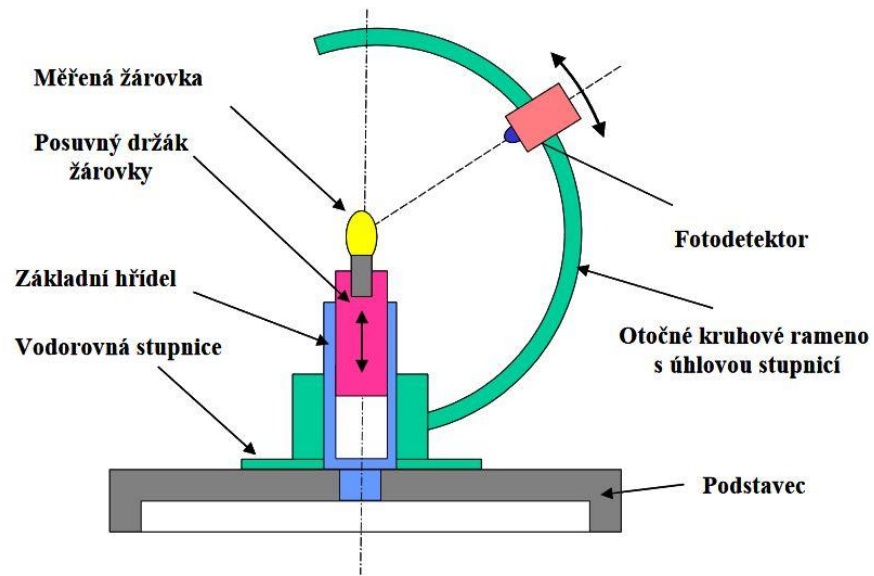
```

fid = fopen('Data_3D.txt');           % Nacteni datoveho souboru
data_3D = fscanf(fid, '%f', [3,inf]); % Pravidla pro tvorbu matice
fclose(fid);                          % Konec prikazu

```

Grafy 3D

- Příprava na úlohu D
- Měření fotometrických parametrů světelných zdrojů



- Načtení datové sady horizontálního úhlu φ z *.txt souboru
- Sloupcový vektor 13×1
- Rozsah 0 až 360 °

%% Nacteni datoveho souboru horizontalniho uhlu

```
fid = fopen('Data_Phi.txt');
Phi = fscanf(fid, '%f', [1, inf]);
fclose(fid);
```

```
% Nacteni datoveho souboru uhlu Phi
% Pravidla pro tvorbu matice
% Konec prikazu
```


- Načtení datové sady vertikálního úhlu θ z *.txt souboru
- Sloupcový vektor 13×1
- Rozsah 0 až 120 °

%% Nacteni datoveho souboru svisleho uhlu

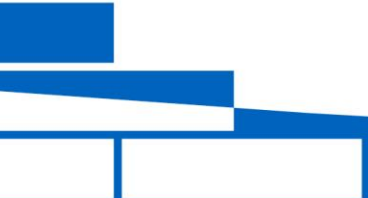
```
fid = fopen('Data_Theta.txt');
Theta = fscanf(fid, '%f', [1, inf]);
fclose(fid);
```

```
% Nacteni datoveho souboru uhlu Theta
% Pravidla pro tvorbu matice
% Konec prikazu
```

- Načtení datové sady signálu světla z *.txt souboru
- Sloupcový vektor 169×1
- Tvorba matice 13×13

%% Nacteni datoveho souboru signalu intenzity svetla

```
fid = fopen('Data_Sig_2.txt'); % Nacteni datoveho souboru signalu svetla
Sig = fscanf(fid, '%f', [13, inf]); % Pravidla pro tvorbu matice
fclose(fid); % Konec prikazu
```



- Normování signálu světla
- Výpočet maxima každého sloupce
- Výpočet totálního maxima
- Max = 1

```
maximum = max (Sig);
```

```
total_maximum = max (maximum);
```

```
Sig_n = Sig*(1/total_maximum);
```

```
% Nalezeni maxima z prvku matice Sig
```

```
% Nalezeni totalniho maxima
```

```
% Normalizace Sig
```

- Tvorba sítě / matice ze sloupcového vektoru
- Vektor 13×1 převedeme na matici 13×13
- Řádky se opakují

`[Theta_m,Phi_m]=meshgrid(Theta,Phi); % Tvorba matice 13x13 ze sloupcoveho vektoru`

- Převod sférických souřadnic na kartézské

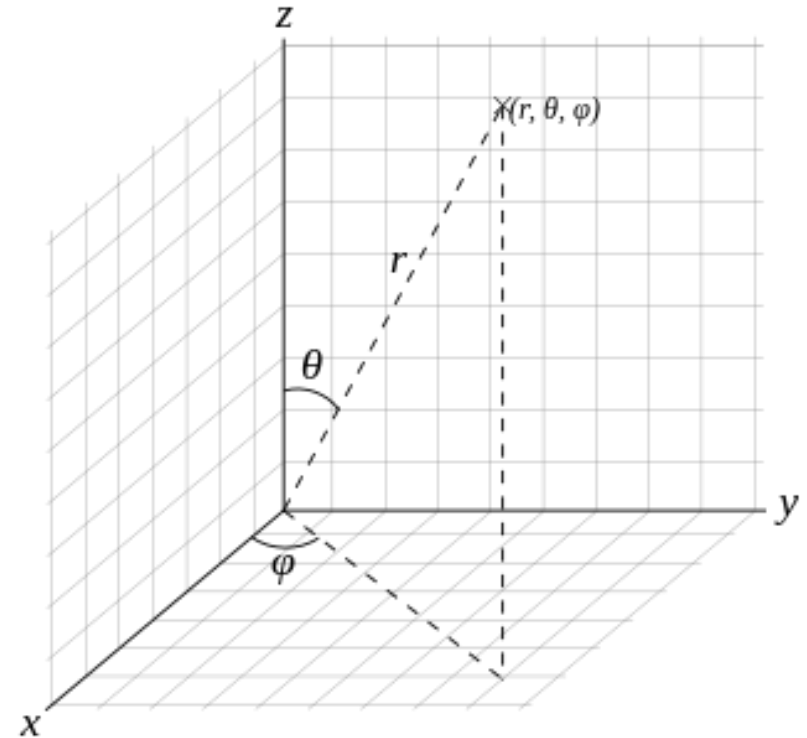
$$x = r \sin \theta \cos \varphi$$

$$y = r \sin \theta \sin \varphi$$

$$z = r \cos \theta$$

% Prevod sferickych souradnic na kartezske

```
x = Sig_n .* sind(Theta_m) .* cosd(Phi_m);
y = Sig_n .* sind(Theta_m) .* sind(Phi_m);
z = Sig_n .* cosd(Theta_m);
```

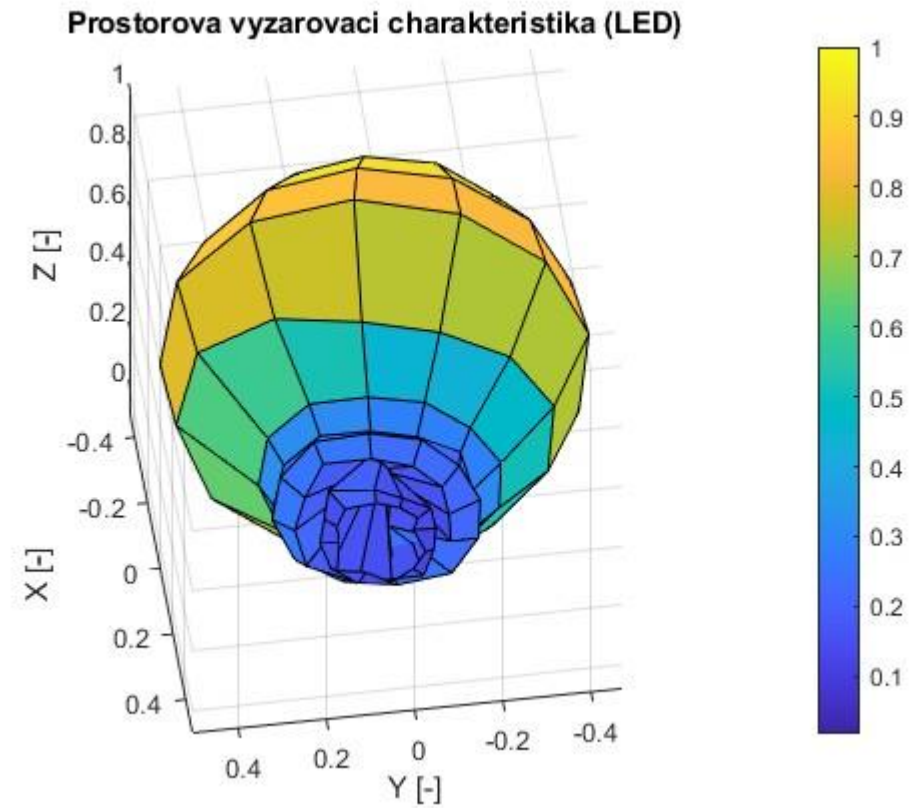
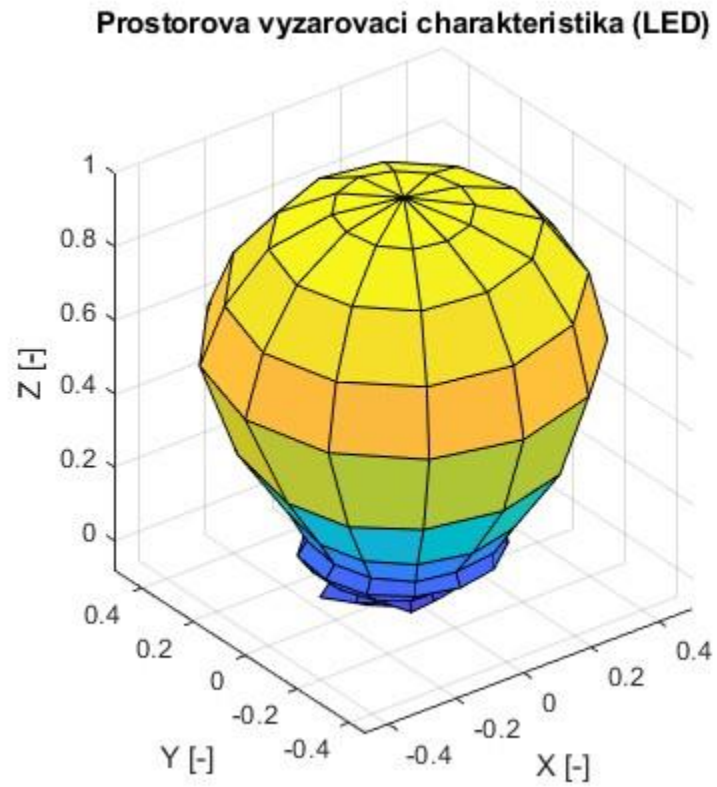


- Konstrukce grafu

%% Konstrukce grafu

```
figure(1)
surf(x,y,z,Sig_n)
title('Prostorova vyzarovaci charakteristika (LED)')
colorbar
axis equal;
xlabel('X [-]')
ylabel('Y [-]')
zlabel('Z [-]')
```

- Konstrukce grafu





**FAKULTA
STAVEBNÍ
ČVUT V PRAZE**

Původní verzi prezentace připravil doc. Ing. Petr Pokorný, Ph.D.

