

## Úloha 1 – Úvod do Matlabu

### Zadání:

Vytvořte v Matlabu:

- funkci `fib_arr(array)`, která pro všechny prvky ve vektoru `array` spočítá fibonacciho hodnotu a vrátí pole těchto hodnot.
- funkci `stat_props(filename)`, která načte zadaný soubor ve formátu CSV a pro každý řádek hodnot vypíše maximum, minimum, medián, průměr, rozptyl, stření hodnotu.

Použití všech funkcí v Matlabu je dovoleno.

### Použité nástroje:

Nástroj Matlab – využita doporučená multilicence ČVUT z odkazu [download.cvut.cz](http://download.cvut.cz).

### 1. Funkce `fib_arr(array)`

Jako Fibonacciho posloupnost je v matematice označována nekonečná posloupnost přirozených čísel, začínající 0, 1, 1, 2, 3, 5, 8, 13, 21, ..., kde každé číslo je součtem dvou předchozích. Čísla nacházející se ve Fibonacciho posloupnosti jsou někdy nazývána Fibonacciho čísla.

Rekurzivní definice Fibonacciho posloupnosti tedy je:

```
fib(n) = 0                pro n = 0
fib(n) = 1                pro n = 1
fib(n) = fib(n - 1) + fib(n - 2)  pro n > 1
```

V prvním kroku jsem napsala funkci `fib` pro výpočet Fibonacciho čísla z libovolného čísla, které je zadáno jako parametr.

The screenshot shows the MATLAB Editor window with the following code in the `fib.m` file:

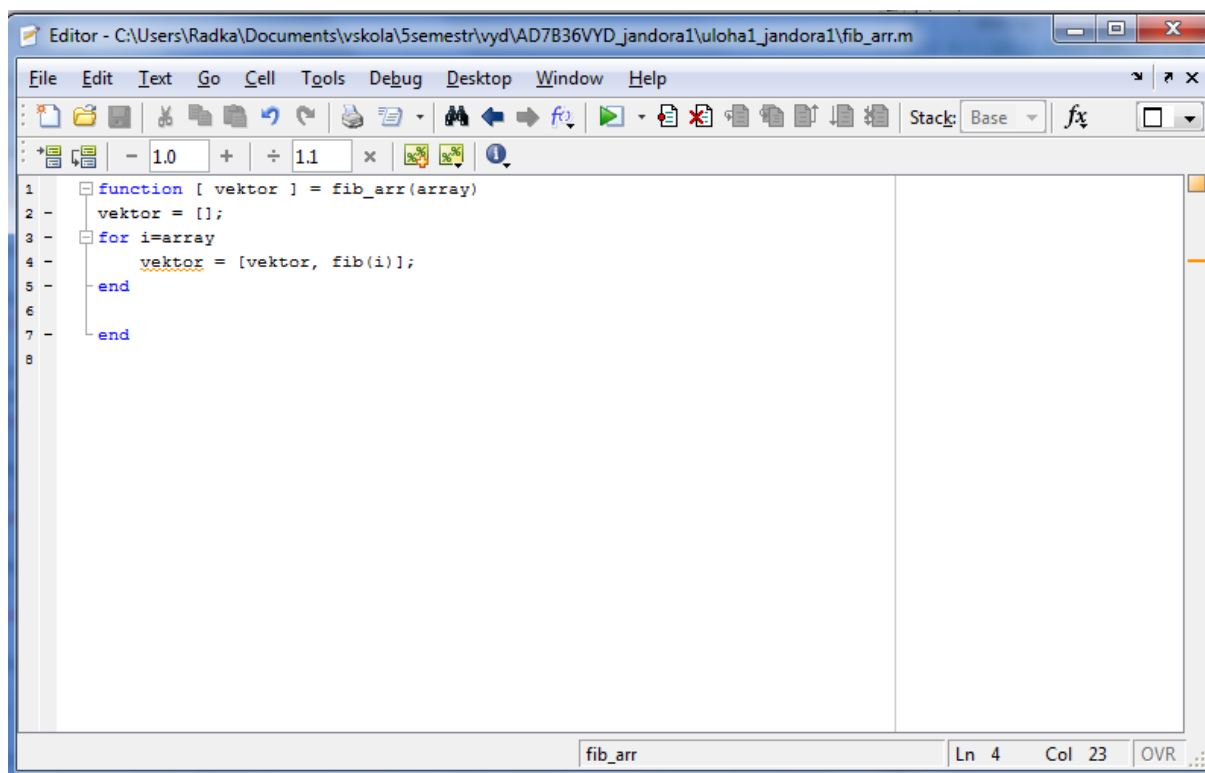
```
1 function fib = fibonacci(cislo)
2     if cislo == 0
3         fib = 0;
4     end
5
6     if cislo == 1
7         fib = 1;
8     end
9
10    if cislo > 1
11        fib = fibonacci(cislo-1) + fibonacci(cislo-2);
12    end
13
14 end
```

The status bar at the bottom indicates the file is `fibonacci`, the cursor is at line 15, column 1, and the overall view is `OVR`.

Příklad:

```
>> fib(8)
ans =
21
```

Ve druhém kroku jsem napsala funkci `fib_array`, která vypočte Fibonacciho čísla z vektoru (= jednorozměrného pole), zadaného jako parametr, a vrátí je jako vektor.



Příklad: 

```
>> fib_arr([0,1,2,3,4,8])
ans =
0      1      1      2      3     21
```

## 2. Funkce `stat_props(filename)`

- **maximum** = ze zadaného vektoru dat vybere největší hodnotu.
- **minimum** = ze zadaného vektoru dat vybere nejmenší hodnotu.
- **medián** = ze zadaného vektoru dat vybere střední hodnotu.
- **průměr** = ze zadaného vektoru dat vrátí aritmetický průměr hodnot
- **rozptyl** = ze zadaného vektoru dat vrátí rozptyl hodnot. Rozptyl je definován jako střední hodnota kvadrátů odchylek od střední hodnoty.
- **modus** = hodnota, která se v daném souboru vyskytuje nejčastěji.

Jako testovací data byl použit soubor `cviceni1.txt` doporučený na stránkách předmětu.

```

1 function [ ] = stat_props( filename )
2     data = csvread(filename);
3     sizeData = size(data);
4     for i = 1:sizeData(1)
5         fprintf('Radek %d\n', i);
6         fprintf('Maximum %f\n', max(data(i,:)));
7         fprintf('Minimum %f\n', min(data(i,:)));
8         fprintf('Median %f\n', median(data(i,:)));
9         fprintf('Prumer %f\n', mean(data(i,:)));
10        fprintf('Rozptyl %f\n', var(data(i,:)));
11        fprintf('Modus %f\n', mode(data(i,:)));
12        fprintf('\n');
13    end
14
15 end
16

```

Výsledek – výpis z Matlabu

```
>> stat_props('cviceni1.txt')
```

<b>Radek 1</b>	<b>Radek 2</b>	<b>Radek 3</b>
Maximum 0.999670	Maximum 9.090400	Maximum 17.924000
Minimum 0.000070	Minimum 1.444200	Minimum -18.578000
Median 0.498330	Median 5.006900	Median 1.066000
Prumer 0.494184	Prumer 4.992351	Prumer 1.054527
Rozptyl 0.083206	Rozptyl 0.943541	Rozptyl 25.415262
Modus 0.024681	Modus 3.810000	Modus -1.265300
 <b>Radek 4</b>	 <b>Radek 5</b>	
Maximum -2.983500	Maximum 3.617900	
Minimum -6.778100	Minimum -3.179000	
Median -5.011050	Median -0.005141	
Prumer -5.005285	Prumer -0.006058	
Rozptyl 0.252080	Rozptyl 0.984277	
Modus -5.033500	Modus -1.415500	

### Zdroje:

[1] *Stránky předmětu A7B36VYD: Vytěžování dat* [online]. Poslední aktualizace neuvedena [cit. 2011-11-25]. Dostupné z WWW <<http://cw.felk.cvut.cz/doku.php/courses/a7b36vyd/cviceni/start>>.

[2] *Wikipedia, Fibonacciho posloupnost* [online]. Poslední aktualizace 21. 11. 2011 [cit. 2011-11-25]. Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Fibonacciho\\_posloupnost](http://cs.wikipedia.org/wiki/Fibonacciho_posloupnost)>.

[3] *Fibonacciho posloupnost* [online]. Poslední aktualizace neuvedena [cit. 2011-11-25]. Dostupné z WWW: <<http://www.algorithm.net/article/116/Fibonacciho-posloupnost>>.