

# **Flexia a torzia v prostredí Mathematica**

## Curvature and torsion in Mathematica

Stanislav Ďuriš



Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

akademický rok: 2014/2015

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Stanislav Ďuriš**

Osobní číslo: **A11597**

Studijní program: **B3902 Inženýrská informatika**

Studijní obor: **Informační a řídicí technologie**

Forma studia: **prezenční**

Téma práce: **Flexe a torze v prostředí Mathematica**

Téma anglicky: **Curvature and Torsion in the Mathematica Environment**

Zásady pro vypracování:

1. Krátce uveďte základní pojmy diferenciální geometrie.
2. Uveďte a popište potřebné příkazy programu Mathematica.
3. Vytvořte interaktivní aplikaci, jejímž vstupem bude uživatelem parametricky zadaná prostorová křivka a výstupem a) vypočítaná její flexe a torze, b) zobrazená křivka a průvodní trojhran (Frenetův repér) postupně v každém bodě dané křivky.
4. Pro popularizaci nahraďte repér nějakým objektem (letadlo, vozík horské dráhy apod.)
5. Vyhledejte na webu applet podobný s Vámi vytvořenou aplikací a porovnejte je.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: tištěná/elektronická

Seznam odborné literatury:

1. GRAY, A. a kol. Modern Differential Geometry of Curves and Surfaces with Mathematica. Boca Raton : Chapman&Hall/CRC, 2006. ISBN 1-58488-448-7
2. KLUVÁNEK I., MIŠÍK L., ŠVEC M. Matematika I; Bratislava, SVTL, 1959
3. CHRAMCOV B. Základy práce v prostředí Mathematica; Zlín, UTB FAI, 2006. ISBN 80-7318-510-5
4. BUDINSKÝ B., KEPR B. Základy diferenciální geometrie s technickými aplikacemi; Praha, SNTL, 1970
5. KOHOUT V; Diferenciální geometrie; Praha, SNTL, 1971
6. Wolfram MathWorld [online]. 2013 [cit. 2015-02-06]. Dostupné z WWW: [<http://mathworld.wolfram.com/Curvature.html>].

Vedoucí bakalářské práce: RNDr. Martin Fajkus, Ph.D.

Ústav matematiky

Datum zadání bakalářské práce: 6. března 2015

Termín odevzdání bakalářské práce: 22. května 2015

Ve Zlíně dne 6. března 2015



doc. Mgr. Milan Adámek, Ph.D.  
děkan

L.S.

prof. Ing. Vladimír Vašek, CSc.  
ředitel ústavu

**Prohlašuji, že**

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

**Prohlašuji,**

- že jsem na diplomové/bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....  
podpis diplomanta

**ABSTRAKT**

Hlavným cieľom bakalárskej práce bolo vytvorenie interaktívnej aplikácie v prostredí Wolfram Mathematica, ktorá je určená predovšetkým pre študentov, ako pomocná aplikácia popri štúdiu matematiky. Obsah práce taktiež obsahuje popis príkazov, ktoré boli využité pri písaní programu.

Klíčová slova: matematika, diferenciálna geometria, Frenetov trojhran, aplikácie, flexia, torzia

## **ABSTRACT**

The main objective of this thesis was to create interactive applications in the environment of Mathematica, which is intended primarily for students as a helper application while studying mathematics. Content of work also includes a description of the commands that were used when writing the program.

Keywords: mathematics, differential geometry, Frenet trihedron, applications, flexion, torsion

Týmto by som chcel poďakovať môjmu vedúcemu bakalárskej práce pánovi RNDr. Martinovi Fajkusovi, Ph.D. za odbornú pomoc a pravidelné konzultácie počas písania tohto diela.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

**OBSAH**

<b>ÚVOD .....</b>	<b>9</b>
<b>TEORETICKÁ ČÁST .....</b>	<b>10</b>
<b>1 DIFERENCIÁLNÁ GEOMETRIA .....</b>	<b>11</b>
1.1 POJEM KRIVKY .....	11
1.2 CHARAKTERISTIKY KRIVKY .....	13
1.3 SPRIEVODNÝ TROJHRAN .....	14
<b>2 SOFTWARE WOLFRAM MATHEMATICA .....</b>	<b>16</b>
2.1 POPIS PROSTREDIA .....	16
<b>PRAKTICKÁ ČÁST .....</b>	<b>18</b>
<b>3 PROGRAM – KRIVKA S POHYBLIVÝM BODOM .....</b>	<b>19</b>
3.1 POPIS FUNGOVANIA PROGRAMU .....	19
3.2 POPIS OVLÁDANIA A VÝSTUPU PROGRAMU .....	21
3.3 POUŽITÉ PRÍKAZY V PROGRAME .....	23
<b>4 POROVNANIE APLIKÁCIE .....</b>	<b>26</b>
<b>ZÁVĚR.....</b>	<b>30</b>
<b>ZÁVĚR V ANGLIČTINE .....</b>	<b>31</b>
<b>SEZNAM POUŽITÉ LITERATURY .....</b>	<b>32</b>
<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK .....</b>	<b>33</b>
<b>SEZNAM OBRÁZKŮ.....</b>	<b>34</b>
<b>SEZNAM TABULEK .....</b>	<b>35</b>
<b>SEZNAM PŘÍLOH .....</b>	<b>36</b>

## ÚVOD

Ako už z názvu vyplýva, bakalárska práca je zameraná na diferenciálnu geometriu a delí sa na teoretickú a praktickú časť.

Teoretická časť je venovaná predovšetkým krivkám, ich charakteristikám. Ďalej tu stručne predstavím vývojové prostredie Wolfram Mathematica 10, ktoré bolo využité pri tvorbe programu pre výpočet krivosti krivky v priestore. Prostredie Wolfram vďaka svojej prepracovanej nápovede výrazne napomáha pri jednoduchom programovaní a je vhodné aj pre programátorov začiatočníkov.

Praktická časť je zameraná na samotný program kde nájdeme popis zdrojového kódu, jednotlivých funkcií, popis ovládania a použité príkazy pri písaní programu. Ďalej praktická časť obsahuje porovnanie funkčnosti vytvoreného programu s webovou aplikáciou.

Táto práca je určená ako podporný materiál pre študentov ktorí by si chceli overiť vypočítané výsledky, prípadne si graficky znázorniť zadanú priestorovú krivku.

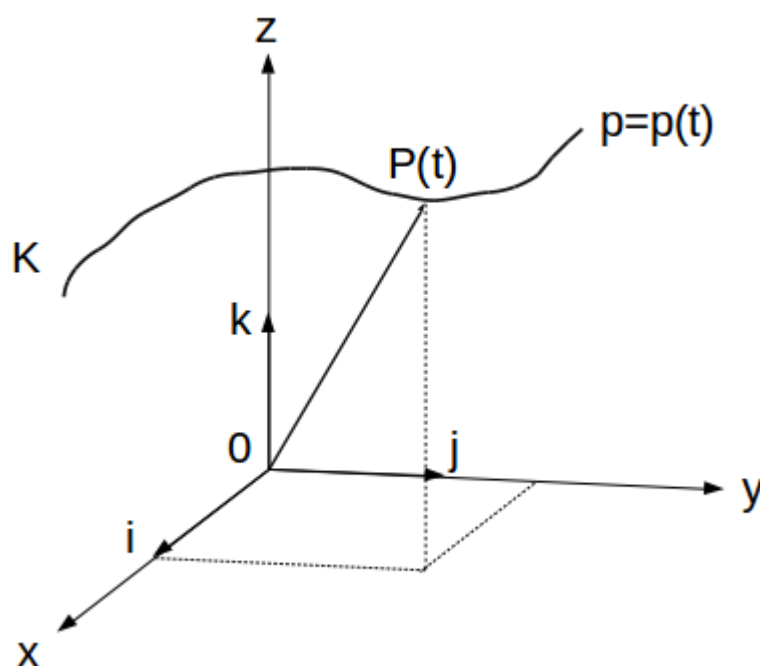
## **I. TEORETICKÁ ČÁST**

## DIFERENCIÁLNÁ GEOMETRIA

Diferenciálna geometria je oblasť matematiky, v ktorej sa skúmajú vlastnosti geometrických útvarov (v našom prípade kriviek v priestore) metódami diferenciálneho počtu. Diferenciálna geometria dáva konkrétnu náplň formálnemu matematickému aparátu množstvom aplikácií fyzikálneho rázu napríklad v mechanike, geodézií a kartografií.

### 1.1 POJEM KRIVKY

Krivku  $k$  si môžeme predstaviť ako súvislú dráhu pohybujúceho sa bodu  $P$ . Ak zvolíme karteziánsku súradnicovú sústavu  $(O, i, j, k)$ , poloha bodu  $P = P(t)$  v čase  $t$  je jednoznačne daná polohovým vektorom  $p(t) = P - O$ .



Obr. č.1: znázornenie krivky v priestore

Základom teórie kriviek je pojem vektorovej funkcie jednej reálnej premennej, funkcií  $p$ , ktorá každému číslu  $t$  z jednorozmerného intervalu priradzuje práve jeden vektor  $p(t)$ . Pre túto funkciu používame zápis

$$p = p(t), \quad t \in J$$

alebo po zložkách

$$p = [x(t), y(t), z(t)], \quad t \in J$$

resp.

$$p = [x(t), y(t), z(t)], \quad t \in J$$

kde  $x(t), y(t), z(t)$  sú reálne funkcie definované na spoločnom intervale  $J$ . Pre vektorovú funkciu môžeme vzhľadom na vzťah  $p = x(t)i, y(t)j, z(t)k$  definovať pojmy limita, spojitost' a derivácia pomocou týchto pojmov pre reálnu funkciu reálnej premennej. Vektorová funkcia  $p(t)$  má v bode  $t_0$  limitu  $p_0$

$$\lim_{t \rightarrow t_0} p(t) = p_0$$

ak pre každú postupnosť  $t_n$  takú, že

$$\lim_{t \rightarrow t_0} p(t) = p_0, \quad t_n \neq t_0, \quad t_n \in J$$

príslušná postupnosť funkčných hodnôt  $\{p(t_n)\}$  konverguje k vektoru  $p_0$ .

Vektorová funkcia  $p = x(t)i, y(t)j, z(t)k$  má v bode  $t_0$  limitu práve vtedy, ak v tomto bode majú limitu reálne funkcie  $x(t), y(t), z(t)$ . Vektorová funkcia  $p(t)$  je spojitá v bode  $t_0$  ak

$$\lim_{t \rightarrow t_0} p(t) = p(t_0)$$

Vektorová funkcia  $p = x(t)i, y(t)j, z(t)k$  je spojitá v bode  $t_0$  práve vtedy, ak sú v tomto bode spojité reálne funkcie  $x(t), y(t), z(t)$ . Vektorová funkcia  $p(t)$  má deriváciu  $dp(t)/dt$  v bode  $t_0$  ak existuje limita

$$\left[ \frac{dp(t)}{dt} \right]_{t=t_0} = \lim_{t \rightarrow t_0} \frac{p(t) - p(t_0)}{t - t_0}$$

Vektorová funkcia  $p = x(t)i, y(t)j, z(t)k$  má v bode  $t_0$  deriváciu práve vtedy, ak v tomto bode majú deriváciu reálne funkcie  $x(t), y(t), z(t)$ .

Metódami diferenciálneho počtu je možné skúmať iba tie krivky, ktorých rovnice spĺňajú určité podmienky, týkajúce sa ich derivácií. Okrem spojitosti funkcie

$p(t)$  predpokladáme spojitost' jej prvej derivácie podľa všeobecného parametra  $t$ , ktorú budeme označovať

$$\dot{p}(t) = \frac{dp(t)}{dt} = \left[ \frac{dx(t)}{dt}, \frac{dy(t)}{dt}, \frac{dz(t)}{dt} \right]$$

Bod  $P(t)$  pre  $t \in J$ , v ktorom existuje  $\dot{p}(t)$  a je  $\dot{p}(t) \neq 0$  nazývame regulárny bod krivky.

Ak je

$$\dot{p}(t) \neq 0,$$

respektíve

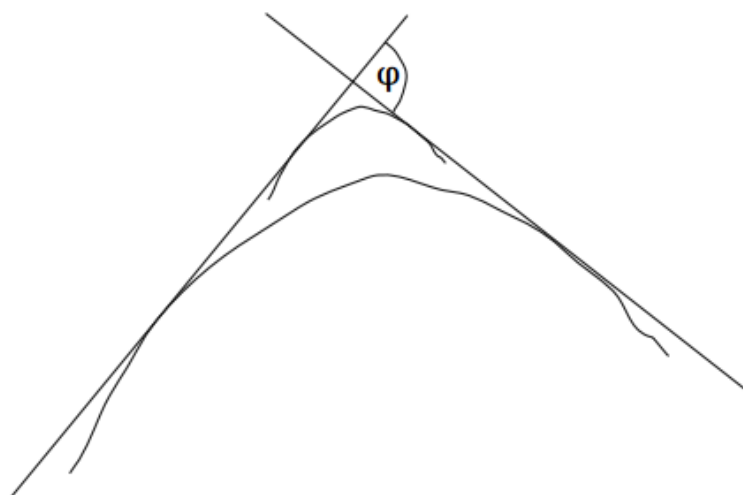
$$(\dot{x}(t))^2 + (\dot{y}(t))^2 + (\dot{z}(t))^2 \neq 0$$

pre ľubovoľný bod  $t \in J$ , krivka  $k$  je regulárna.

## 1.2 CHARAKTERISTIKY KRIVKY

Flexia krivky

Pri pohybe dotykového bodu  $P(t_0)$  po regulárnej krivke sa mení smer dotyčnice v tomto bode. Rýchlosť zmeny smeru dotyčnice charakterizuje stupeň zakrivenia krivky. Čím viac sa v okolí bodu dotyku krivka odkláňa od dotyčnice v tomto bode, tým má väčšiu prvú krivosť (flexiu).



Obr. č.2: krivosť

Ak je krivka  $k$  definovaná vektorovou rovnicou, potom jej krivosť v bode  $P(t_0)$  vypočítame podľa vzťahu

$$K^2(t_0) = \frac{\begin{vmatrix} \dot{y}(t_0) & \dot{z}(t_0) \\ \ddot{y}(t_0) & \ddot{z}(t_0) \end{vmatrix}^2 + \begin{vmatrix} \dot{z}(t_0) & \dot{x}(t_0) \\ \ddot{z}(t_0) & \ddot{x}(t_0) \end{vmatrix}^2 + \begin{vmatrix} \dot{x}(t_0) & \dot{y}(t_0) \\ \ddot{x}(t_0) & \ddot{y}(t_0) \end{vmatrix}^2}{(\dot{x}(t_0)^2 + \dot{y}(t_0)^2 + \dot{z}(t_0)^2)^3}$$

Pri pohybe bodu  $P(t_0)$  po regulárnej krivke sa mení poloha oskulačnej roviny v tomto bode. Čím viac sa v okolí bodu  $P(t_0)$  krivka odchyľuje z oskulačnej roviny v tomto bode, tým má väčšiu druhú krivosť – torziu. Ak je krivka definovaná vektorovou rovnicou, potom jej torziu v bode  $P(t_0)$  môžeme vypočítať podľa vzťahu

$$T(t_0) = \frac{\begin{vmatrix} \dot{x}(t_0) & \dot{y}(t_0) & \dot{z}(t_0) \\ \ddot{x}(t_0) & \ddot{y}(t_0) & \ddot{z}(t_0) \\ \ddot{x}(t_0) & \ddot{y}(t_0) & \ddot{z}(t_0) \end{vmatrix}}{\begin{vmatrix} \dot{y}(t_0) & \dot{z}(t_0) \\ \ddot{y}(t_0) & \ddot{z}(t_0) \end{vmatrix}^2 + \begin{vmatrix} \dot{z}(t_0) & \dot{x}(t_0) \\ \ddot{z}(t_0) & \ddot{x}(t_0) \end{vmatrix}^2 + \begin{vmatrix} \dot{x}(t_0) & \dot{y}(t_0) \\ \ddot{x}(t_0) & \ddot{y}(t_0) \end{vmatrix}^2}$$

### 1.3 SPRIEVODNÝ TROJHRAN

Ku každému bodu  $P(t_0)$  regulárnej krivky  $k$ , v ktorom existuje  $\dot{p}(t_0) \neq 0$ , je možné jednoznačne priradiť tri vektory, ktoré spolu s bodom  $P(t_0)$  určujú pravoúhly trojhran s vrcholom v bode  $P(t_0)$ . Jeho hranami sú dotyčnica  $d$ , hlavná normála  $n$ , a binormála  $b$ . Steny tvoria oskulačnú rovinu  $\tau$ , normálovú rovinu  $\nu$  a rektifikačnú rovinu  $\mu$ . Pri pohybe bodu  $P(t_0)$  po regulárnej krivke sa mení poloha tohto trojhranu. Preto sa trojhran nazýva sprievodný alebo Serretov trojhran. Umožní nám výstižne popísať tvar krivky  $k$  v okolí bodu  $P(t_0)$ .

#### Dotyčnica krivky

Medzi všetkými priamkami, ktoré prechádzajú bodom  $P(t_0)$  regulárnej krivky  $k$  existuje významná priamka, ktorú nazývame dotyčnica. Na krivke  $k$  zvolíme dva rôzne body  $P(t_0)$  a  $P(t_0 + h)$ ,  $h \neq 0$ . Priamka prechádzajúca týmito bodmi sa nazýva sečnica krivky  $k$ . Dotyčnica v bode  $P(t_0)$  je limitnou polohou sečnice pre  $h \rightarrow 0$ . V každom bode  $P(t_0) = [x_0(t), y_0(t), z_0(t)]$  regulárnej krivky existuje práve jedna dotyčnica. Ak je krivka popísaná rovnicou  $p = p(t)$ ,  $t \in J$  prípadne  $x = x(t)$ ,  $y = y(t)$ ,  $z = z(t)$ ,  $t \in J$ , potom vektorová rovnica dotyčnice je  $d = p(t_0) + \lambda \dot{p}(t_0)$  a parametrické rovnice dotyčnice sú  $x = x(t_0) + \lambda \dot{x}(t_0)$ ,  $y = y(t_0) + \lambda \dot{y}(t_0)$ ,  $z = z(t_0) + \lambda \dot{z}(t_0)$  kde  $\lambda \in (-\infty, \infty)$  je parameter,  $d$  je označenie pre polohový vektor ľubovoľného bodu dotyčnice a  $x, y, z$  sú súradnice tohoto vektora, resp. súradnice ľubovoľného bodu dotyčnice.

### Hlavná normála a binormála krivky

Každá priamka prechádzajúca bodom  $P(t_0)$  regulárnej krivky  $k$  kolmá na dotyčnicu sa nazýva normála krivky  $k$  v bode  $P(t_0)$ . Normálu, ktorá leží v oskulačnej rovine budeme nazývať hlavnou normálou a označovať ako  $n$ . Pod pojmom binormála  $b$  krivky  $k$  budeme rozumieť priamku, ktorá prechádza bodom  $P(t_0)$  a je kolmá súčasne na príslušnú dotyčnicu i hlavnú normálu. Binormála je teda kolmá na oskulačnú rovinu a preto vektorová rovnica binormály je  $b = p(t_0) + \lambda(\dot{p}(t_0) \times \ddot{p}(t_0))$  a parametrické rovnice binormály sú

$$\begin{aligned} x &= x(t_0) + \lambda \begin{vmatrix} \dot{y}(t_0) & \dot{z}(t_0) \\ \ddot{y}(t_0) & \ddot{z}(t_0) \end{vmatrix} & y &= y(t_0) + \lambda \begin{vmatrix} \dot{z}(t_0) & \dot{x}(t_0) \\ \ddot{z}(t_0) & \ddot{x}(t_0) \end{vmatrix} & z &= z(t_0) + \lambda \begin{vmatrix} \dot{x}(t_0) & \dot{y}(t_0) \\ \ddot{x}(t_0) & \ddot{y}(t_0) \end{vmatrix} \end{aligned}$$

pre každú hranu sprievodného trojhranu platí vzťah  $b = d \times n$ ,  $d = n \times b$ ,  $n = b \times d$

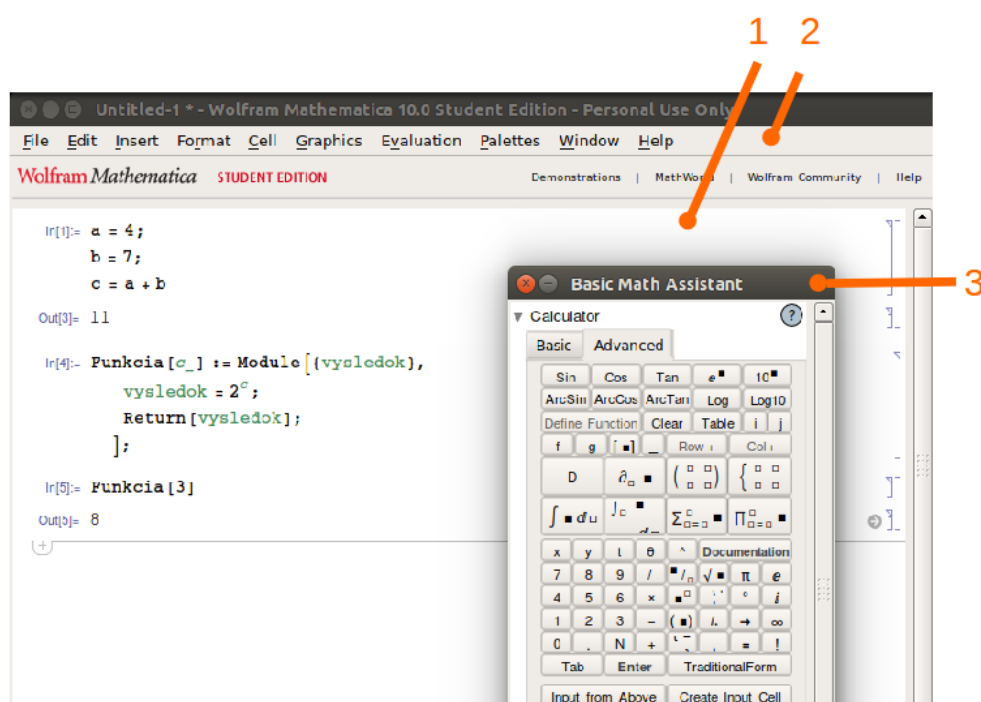
## SOFTWARE WOLFRAM MATHEMATICA

Produkt spoločnosti Wolfram Researches známy ako Mathematica (taktiež Wolfram Mathematica) je pomenovaný po jej zakladateľovi – Stephenovi Wolframovi a v dnešnej dobe patrí



k jedným z najlepších systémov na svete z hľadiska výpočetného výkonu.

Obr. č.3 : Úvodné logo softwaru Wolfram Mathematica



### 2.1 POPIS PROSTREDIA

Obr. č.4 : Popis prostredia Wolfram Mathematica

Na obrázku č. máme spustené prostredie Wolfram Mathematica kde č.1 označuje hlavnú lištu programu, 2. notebook a 3. paleta nástrojov Basic Math Assistant.

Obsah hlavnej lišty:

File – práca s notebookom: nový, otvorenie, uloženie, export ...

Edit – kopírovanie, vkladanie, nastavenia...

Format – design notebooku, nastavenia fontov...

Cell – práca s bunkami: zoskupovanie, delenie...

Graphics – práca s grafickými prvkami

Evaluation – kompilácia a debug zdrojového kódu

Palettes – pre vkladanie matematických vzorcov

Window – vzhľad okna notebooku

Help – nápoveda, dokumentácia...

## **II. PRAKTICKÁ ČÁST**

## PROGRAM – KRIVKA S POHYBLIVÝM BODOM

### 3.1 Popis fungovania programu

Obsah zdrojového kódu tvoria dve funkcie: *Krivka* a *Manipulate*. Samotné jadro programu je uložené vo funkcii *vykresli[]*, kde sú deklarované všetky potrebné matematické vrátane vzorcov pre krivosti krivky. Do funkcie *vykresli[]* je zakomponovaná ďalšia dôležitá funkcia *kreslibod[]* ktorá je určená pre grafickú prácu s pohyblivým bodom a vlastnosťami krivky v danom bode.

#### Funkcie programu:

- *Krivka[]* - hlavná časť programu, inicializácia vstupu

```
Krivka[K_] := Module[{},  
  tmin = 0;  
  tmax = 6  $\pi$ ;  
  
  k = K;  
  
  If[k == Null, Return["Zadajte vstupnú krivku a pokračujte klávesou ENTER"]];  
  
  Vykresli[k];  
  
  Grid[{"Vykreslený graf"}, {Vykresli[k]}, Frame -> All]  
];
```

- *Vykresli[]* - grafické vykreslenie grafu zadanej priestorovej krivky a deklarácia matematických výpočtov

```

Vykresli[k_] := Module[{K = k, d1, d2, d3, Vektor},
  d1 = D[Vektor, x];
  d2 = D[Vektor, {x, 2}];
  d3 = D[Vektor, {x, 3}];

  d1x = d1[[1]]; d2x = d2[[1]]; d3x = d3[[1]];
  d1y = d1[[2]]; d2y = d2[[2]]; d3y = d3[[2]];
  d1z = d1[[3]]; d2z = d2[[3]]; d3z = d3[[3]];

  Vektor = Flatten[{K[[1]], K[[2]], K[[3]]}];
  Vektor = DeleteCases[Vektor, 0];
  Vektor = Vektor /. t -> x;

  Dotynical[t_] := D[Vektor, x] /. x -> t;
  Dotynica[t_] := Normalize[Dotynical[t]] /. x -> t;
  Normalal[t_] := D[Vektor, {x, 2}] /. x -> t;
  Normala[t_] := Normalize[Normalal[t]] /. x -> t;
  Binormala[t_] := (Dotynica[t] * Normala[t]);
  BOD[t_] := Vektor /. x -> t;

  KresliBod[t_, dot_, norm_, binorm_, Letadlo_, Bod_]

  Graf = ParametricPlot3D[BOD[t], {t, tmin, tmax}, PlotStyle -> Darker, PlotRangePadding -> 1, BoxRatios -> {10, 10, 10}, AxesOrigin -> {0, 0, 0},
    AxesLabel -> {"x", "y", "z"}, BoxStyle -> Directive[Dashed, White, Thick], ImageSize -> Medium];

  Flex[t_] := Simplify[
$$\frac{\sqrt{(d1x * d2y - d2x * d1y)^2 + (d1x * d2z - d2x * d1z)^2 + (d1y * d2z - d2y * d1z)^2}}{(d1x^2 + d1y^2 + d1z^2)^{3/2}}$$
] /. x -> t;

  Torz[t_] := Simplify[Abs[
$$\begin{vmatrix} d1x & d1y & d1z \\ d2x & d2y & d2z \\ d3x & d3y & d3z \end{vmatrix}$$
] / ((d1x * d2y - d2x * d1y)^2 + (d1x * d2z - d2x * d1z)^2 + (d1y * d2z - d2y * d1z)^2) / N] /. x -> t;

  KresliBod[t_, dot_, norm_, binorm_, Letadlo_, Bod_] := Module[{},

  Show[
    Graphics3D[{Opacity[dot], Red,
      Arrowheads[0.03],
      Arrow[{BOD[t], BOD[t] + Dotynica[t]}] (*DOTYCNICA*)
    }],
    Graphics3D[{Opacity[norm], Green,
      Arrowheads[0.03],
      Arrow[{BOD[t], BOD[t] - Normala[t]}] (*NORMALA*)
    }],
    Graphics3D[{Opacity[binorm], Blue,
      Arrowheads[0.03],
      Arrow[{BOD[t], BOD[t] - Binormala[t]}] (*BINORMALA*)
    }],
    Graphics3D[{Opacity[Bod], AbsolutePointSize[15], Orange, Point[BOD[t]] (*BOD*)
    }],
    Graphics3D[{Opacity[Letadlo], Purple, Thick, (*Letadlo*)
      Line[{
        (*Letadlo*)
        {BOD[t] + Dotynica[t], BOD[t] - 0.5 * Normala[t], BOD[t] + 0.5 * Normala[t], BOD[t] + Dotynica[t], BOD[t],
          BOD[t] - 0.5 * Binormala[t], BOD[t] + 0.5 * Dotynica[t]}
      }]}
  ]];

  Graf = ParametricPlot3D[BOD[t], {t, tmin, tmax}, PlotStyle -> Darker, PlotRangePadding -> 1, BoxRatios -> {10, 10, 10},
    AxesOrigin -> {0, 0, 0}, AxesLabel -> {"x", "y", "z"}, BoxStyle -> Directive[Dashed, White, Thick], ImageSize -> Medium];

```

- KresliBod[] - grafické vykreslenie bodu krivky spolu s vlastnosťami krivky v zadanom bode

- Manipulate[] - ovládanie a nastavenie hodnôt funkcie KresliBod[] a výpis vypočítaných hodnôt

```

Manipulate[
  Show[Graf, KresliBod[t, dot, norm, binorm, Letadlo, Bod]],

  Style["Zadaná krivka: " Dynamic[BOD[t]], 13, Bold],
  Delimiter,
  Control[{{t, tmin, "t:"}, tmin, tmax, 0.001}],

  (*{{ZadajKrivku,"Zadana krivka: "}},*)

  {{Bod, 1, "Bod"}, {1, 0}},
  {{dot, 0, "Dotyčnica"}, {1, 0}},
  {{norm, 0, "Normála"}, {1, 0}},
  {{binorm, 0, "Binormála"}, {1, 0}}, Delimiter,
  {{Letadlo, 0, "LETADLO"}, {1, 0}},

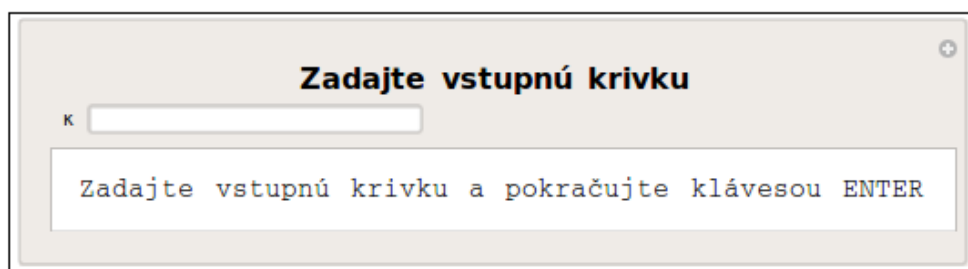
  Delimiter,
  Style["Suradnice bodu: " Dynamic[BOD[t]], 13, Italic],
  Delimiter,
  Style["Flexia krivky: " Dynamic[Flex[t]], 13, Bold],
  Delimiter,
  Style["Torzia krivky: " Dynamic[Torz[t] // MatrixForm], 13, Bold],
  ControlPlacement → Right, FrameLabel → {"", "", Style["Krivka s pohyblivým bodom", 15, Bold]]]

Grid[{{Manipulate[Krivka[k], {k, Null}, ControlPlacement → Top,
  FrameLabel → {"", "", Style["Zadajte vstupnú krivku", 15, Bold]}]}},
  Frame → All];

```

- Manipulate[] - zadanie hodnoty pre funkciu Krivka[]

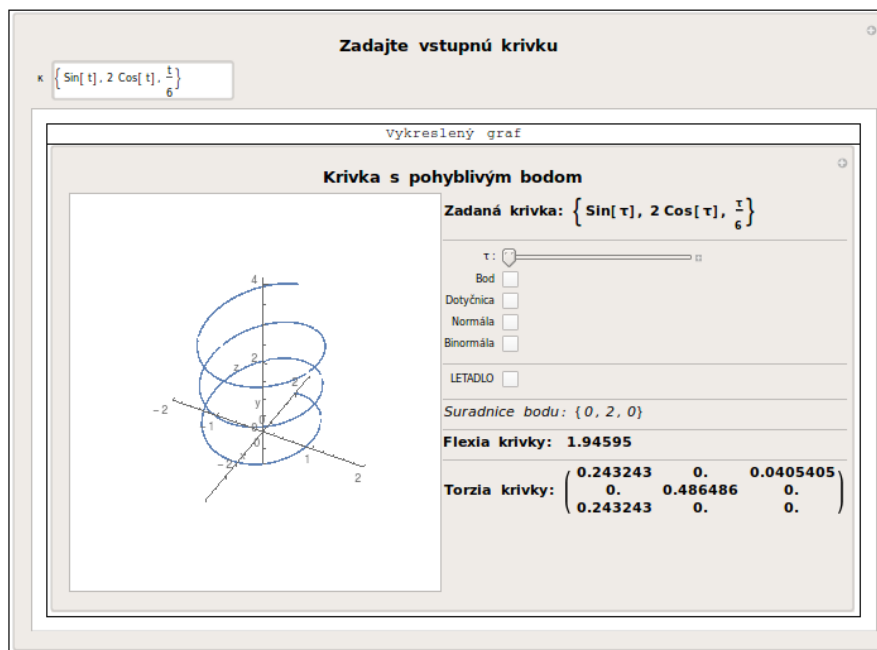
### 3.2 Popis ovládania a výstupu programu



Po spustení programu sa objaví okno, ktoré užívateľ a vyzve k zadaniu vstupnej krivky.

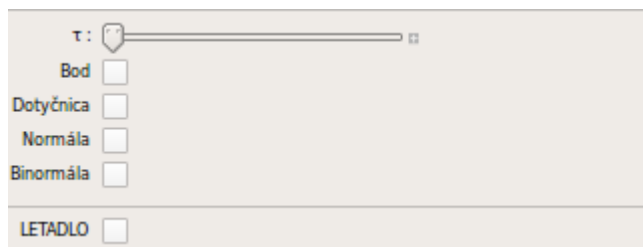
*Obr. č. 5: úvodné okno programu*

Po zadaní krivky a následnom potvrdení sa výstup programu zmení:



Obr. č. 6: výstup programu

Program je v tomto momente usporiadaný do dvoch častí. V ľavej časti sa nachádza samotný výstup aplikácie - graf priestorovej krivky. Pravá časť okna je voľne nastavovateľná (možnosti zobrazenia bodu, dotyčnice, normály a binormály v bode) a zároveň slúži ako výstup dynamických podfunkcií resp. prvej a druhej krivosti krivky.

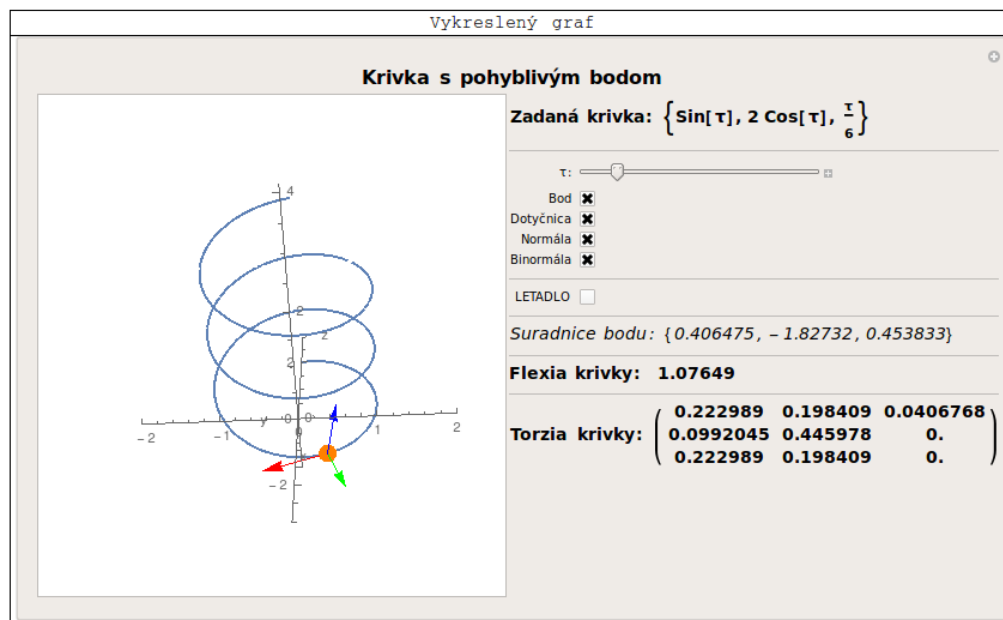


Obr. č. 7: nastavenia vlastností bodu

### Nastavenia vlastností bodu

- checkbox
  - bod
  - dotyčnica
  - normála
  - binormála
- manipulator
  - **T**

Ak je vybraný niektorý z vyššie uvedených checkboxov, vybraný objekt sa zobrazí v ľavej časti – obrázok č.8. Manipulator nám slúži k pohybu bodu krivky. Jeho maximálny rozsah je daný dĺžkou



krivky.

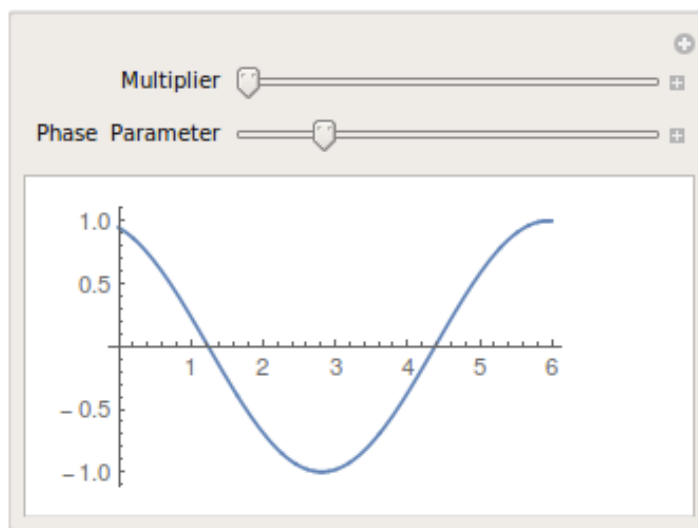
Obr. č. 8: Vybrané hodnoty

### 3.3 Použité príkazy v programe

*Manipulate[]:*

- príkaz slúžiaci pre zakomponovanie rôznych ovládacích grafických prvkov do vytvorenej funkcie.
- Príklad: využitie príkazu `manipulate` s využitím dvoch posuvníkov ovládajúcich hodnotu amplitúdy a fáze:

```
Manipulate[Plot[Sin[a x + b], {x, 0, 6}],  
  {{a, 2, "Multiplier"}, 1, 4},  
  {{b, 0, "Phase Parameter"}, 0, 10}]
```

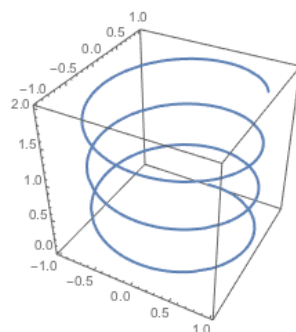


Obr. č. 9: příkaz Manipulate[]

*ParametricPlot3D[]:*

- příkaz sloužící pro grafické vykreslení prostorových křivek

```
ParametricPlot3D[{Sin[u], Cos[u], u/10}, {u, 0, 20}]
```



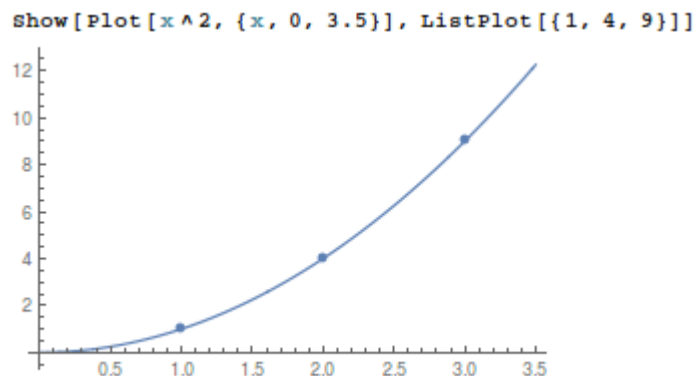
- příklad: grafické vykreslení parametricky zadanej křivky:

Obr. č. 10: příkaz ParametricPlot3D[]

*Show[]:*

- příkaz sloužící pro zakomponování dvou nebo více funkcí do společného grafického výstupu

- príklad: vykreslenie krivky a bodov do spoločnej výstupnej bunky:



Obr. č. 11: príkaz Show[]

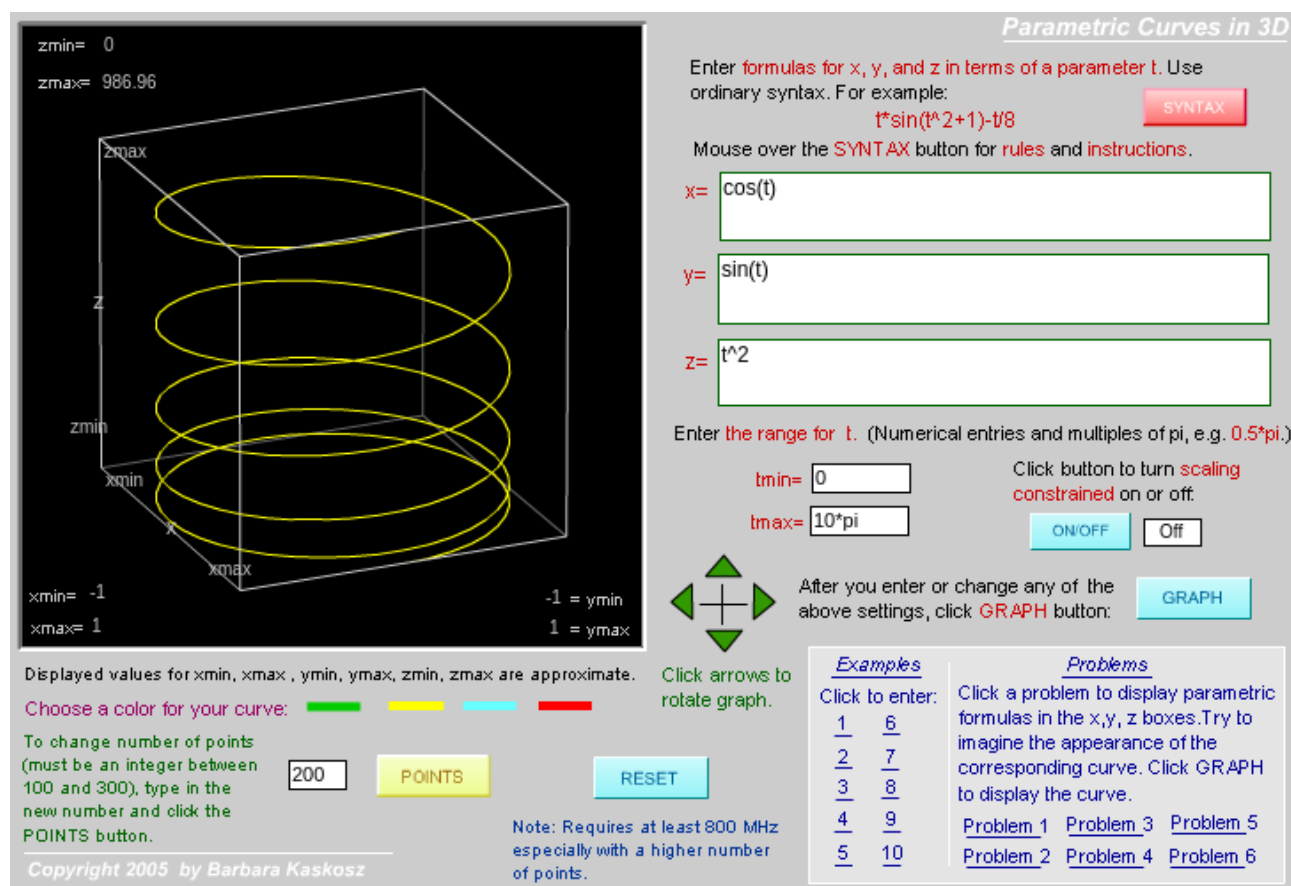
Ostatné použité príkazy:

- *Arrow[]* - vykreslí šípku, vstupom je počiatočný a konečný bod.
- *Control[]* - určený pre zobrazenie grafického ovládacieho prvku.
- *D[]* - zderivuje funkciu podľa zvolenej premennej.
- *DeleteCases[]* - príkaz pre vymazanie prvkov z vektoru.
- *Dynamic[]* - slúži pre dynamické obnovovanie hodnoty zadanej premennej.
- *Flatten[]* - zjednotenie viacerých vektorov do jedného.
- *Graphics3D[]* - určený pre prácu s 3D grafickými prvkami.
- *Grid[]* - slúži pre formátovanie zadaného výrazu.
- *If[podmienka,výraz]* – podmienkový príkaz.
- *Normalize[]* - príkaz pre normovanie výsledku.
- *Opacity[]* - nepriehľadnosť, využívaná v grafike.
- *Return[]* - vráti návratovú hodnotu funkcie.
- *Simplify[]* - vykoná sekvenciu algebraických transformácií a vráti najjednoduchšiu formu.
- *Style[]* - formátovacia funkcia, upraví design výsledku.



## POROVNANIE APLIKÁCIE

Prostredníctvom internetu som vyhľadal aplikáciu, ktorá plní rovnakú funkciu ako mnou naprogramovaná aplikácia a porovnám ich v niekoľkých kategóriách. Označenie „\*“ symbolizuje víťaza kategórie.



Obr. č. 12: webová aplikácia

### 1. Design aplikácie.

Zmysel pre design býva u užívateľov rôzny a preto sa ťažko hodnotí, ktorá aplikácia vypadá na pohľad lepšie a má všetko presne tam, kde to užívateľ intuitívne nájde. Na obrázku č. je design webovej aplikácie. Posúdte sami. V tejto kategórii nevyhráva žiadna aplikácia.

Hodnotenie:

1.	webová aplikácia	naprogramovaná aplikácia
design aplikácie	*	*

Tab.č.1: design

### 2. Volba vstupu.

V tejto kategórii jednoznačne zvíťazila moja aplikácia. Z užívateľského hľadiska to znamená pohodlnejšie zadávanie vstupu v podobe vektora, ktorý program spracuje. Zato u webovej aplikácie je so zadávaním vstupných hodnôt miestami problém. Program nedokáže rozoznať zadaný vstup v

podobe „2t“ ale všetko treba napísať úplne – teda „2\*t“ apod. Jediným plusom je niekoľko nadefinovaných funkcií, ktoré je možné vybrať zo zoznamu.

Hodnotenie:

2.	webová aplikácia	naprogramovaná aplikácia
zadávanie vstupu		*

Tab.č.2: Volba vstupu

### 3. Ovládanie aplikácie.

Na strane webovej aplikácie je dosť nešikovné. S výstupom sa manipuluje nepohodlne – za pomoci štyroch navigačných tlačítok umiestnených vedľa grafu. V prípade naprogramovanej aplikácie, ovládanie výstupu prebieha prostredníctvom natočenia grafu kurzorom. Víťazom sa znovu stáva naprogramovaná aplikácia, aj keď hlavnú zásluhu na tom má vývojové prostredie.

Hodnotenie:

3.	webová aplikácia	naprogramovaná aplikácia
ovládanie aplikácie		*

Tab.č.3: Ovládanie aplikácie

### 4. Výpočty.

Webová aplikácia – minimá a maximá funkcie v osách sústavy, naprogramovaná aplikácia – zobrazené krivosti.

Hodnotenie:

4.	webová aplikácia	naprogramovaná aplikácia
výpočty		*

Tab.č.4: Výpočty

### 5. Ovládanie výstupu.

Výstupom naprogramovanej aplikácie je graf krivky, bod a výpočty krivostí v danom bode. Vstupom webovej aplikácie je graf funkcie. Víťaz – naprogramovaná aplikácia.

Hodnotenie:

5.	webová aplikácia	naprogramovaná aplikácia
ovládanie výstupu		*

Tab.č.5: Ovládanie výstupu

**Zhodnotenie a záver:**

Naprogramovaná aplikácia dopadla v porovnaní s webovou aplikáciou vo všetkých kategóriách lepšie. Velkú zásluhu na tom má vývojové prostredie Wolfram, ktoré je konštruované pre kombinácie matematiky, programovania a 3D grafiky.

## ZÁVĚR

Bakalárska práca je rozdelená na časti teoretickú (vysvetlenie základných pojmov z oblasti diferenciálnej geometrie) a praktickú (zameraná na vytvorený program).

Hlavným výsledkom tejto práce je interaktívny program, ktorého ovládanie prípadne nastavenia, nie je nijak zložitý. Užívateľ si zvolí parametricky zadanú krivku, zadá hodnotu a po potvrdení vidí výsledný graf priestorovej krivky. Program by sa dal ďalej ošetriť proti nesprávnemu zadávaniu vstupného vektoru, za využitia podmienkových cyklov, ktoré by ho mohli viac prispôbiť užívateľovi.

V praktickej časti je vysvetlenie jednotlivých dôležitých príkazov využitých počas písania programu. Oceniť by to mohli hlavne študenti, ktorí nemajú nejak moc skúseností so softwarom of spoločnosti Wolfram .

## ZÁVĚR V ANGLIČTINĚ

this thesis is divided into theoretical (explanation of basic terms of differential geometry) and practical (focused on a creating a program ) part.

The main result of this work is an interactive program which the control or setting, is not difficult. The user selects a parameter set curve, set up a value and after confirmation he can see the resulting graph spatial curve. The program could be further treated against incorrect entering the input vector for use of conditional cycles which would be better adapted to the user.

In the practical part is the explanation of the major important commands used throughout the writing program. It should be appreciate by students who do not have too much experience with the software of the company Wolfram.

## SEZNAM POUŽITÉ LITERATURY

- [Ошибка! Не указана последовательность.] CHRAMCOV, Bronislav, Základy práce v prostředí Mathematica. Zlín: Univerzita Tomáše Bati, 2006. ISBN 80-510-5.
- [2] GRAY, A. a kol. Modern Differential Geometry of Curves and Surfaces with Mathematica Boca Ration: Chapman&Hall/CRC, 2006, ISBN 1-58488-448-7
- [3] KULVÁNEK I., MIŠÍK L., ŠVEC M. Matematika I; Bratislava, SVTL, 1959
- [4] BUDINSKÝ B., KEPR B. Základy diferenciální geometrie s technickými aplikacemi; Praha, SNTL, 1970
- [5] KOHOUT V; Diferenciální geometrie; Praha, SNTL, 1971
- [6] Wolfram MathWorld [online]. 2013 [cit.2015-02-06]. Dostupné z WWW: [http://mathworld.wolfram.com/Curvature.html]

## SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

## SEZNAM OBRÁZKŮ

Obr. č.1: znázornenie krivky v priestore

*Obr. č.2: krivosť*

*Obr. č.3 : Úvodné logo softwaru Wolfram Mathematica*

*Obr. č.4 : Popis prostredia Wolfram Mathematica*

*Obr. č. 5: úvodné okno programu*

*Obr. č. 6: výstup programu*

*Obr. č. 7: nastavenia vlastností bodu*

*Obr. č. 8: Vybrané hodnoty*

*Obr. č. 9: príkaz Manipulate[]*

*Obr. č. 10: príkaz ParametricPlot3D[]*

*Obr. č. 11: príkaz Show[]*

*Obr. č. 12: webová aplikácia*

## SEZNAM TABULEK

*Tab.č.1: design*

*Tab.č.2: Volba vstupu*

*Tab.č.3: Ovládanie aplikácie*

*Tab.č.4: Výpočty*

*Tab.č.5: Ovládanie výstupu*

## **SEZNAM PŘÍLOH**

Vytvorený program pre výpočet krivosti priestorovej krivky umiestnený na priloženom CD

**PŘÍLOHA P I: NÁZEV PŘÍLOHY**