

(j)edit

Interaktívne vyšetrovanie priebehu elementárnych funkcií

AUTOR: Juraj Vetrák

ŠKOLITEĽ: Ing. Ján Komara, PhD.

cieľ práce - prehľad

interaktívny editor

edukačný nástroj

matematická analýza

priebeh funkcie

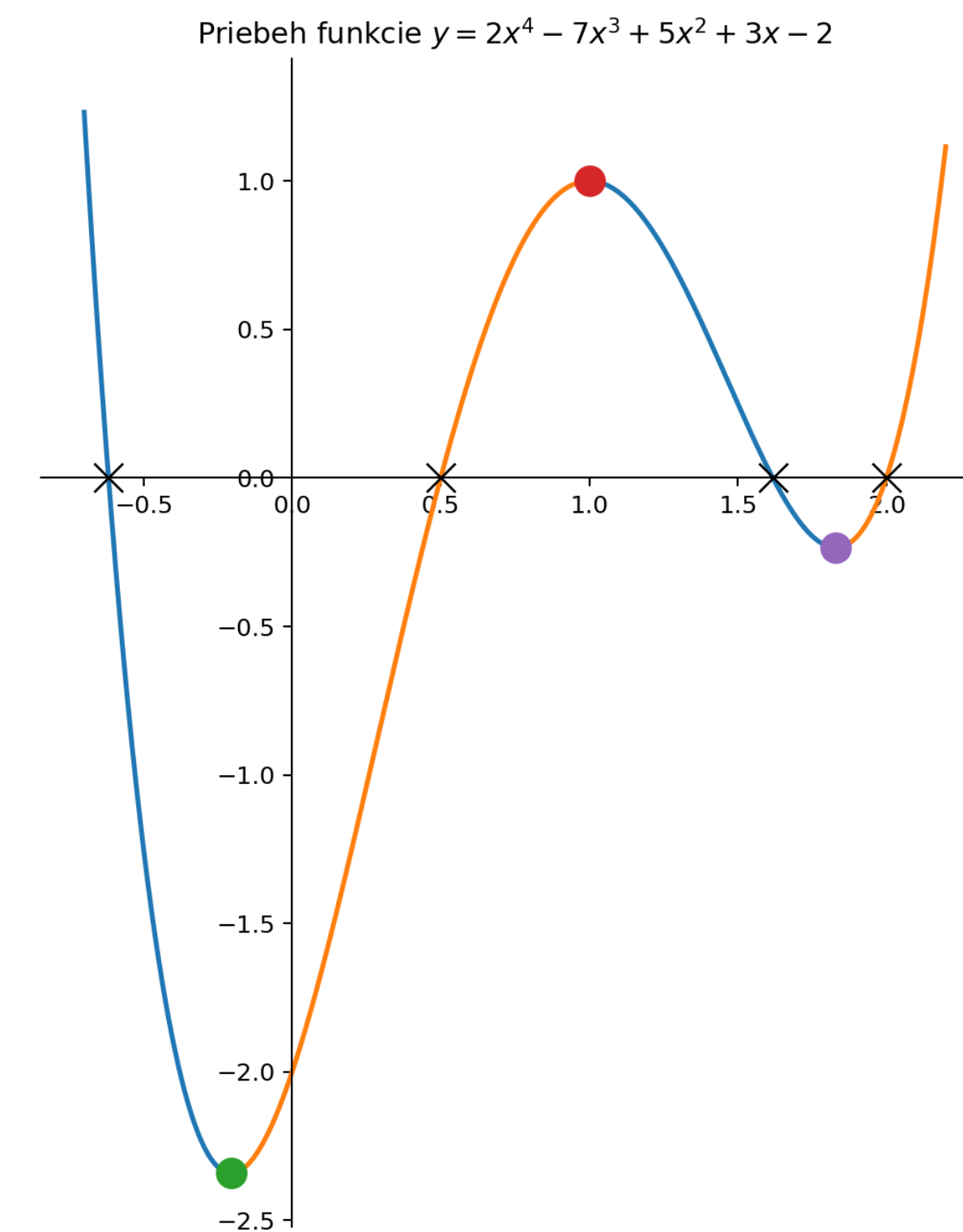
vizualizačná pomôcka

prostredie jupyter notebook

cesta k editoru...

TYPICKÁ SITUÁCIA

- študent úvodného kurzu matematickej analýzy
- predpokladaná neznalosť diferenciálneho počtu
- základná znalosť programovacieho jazyka Python
- vie nakresliť graf funkcie pomocou grafických knižníc
- vie z grafu vyčítať niektoré kvantitatívne charakteristiky danej funkcie



cesta k editoru...

PROBLÉM

- niektoré vlastnosti funkcie poskytol priamo graf
- ako však určiť presnejšie súradnice/intervaly?
- bez diferenciálneho počtu to zrejme jednoduché nebude...
- ...či?



cesta k editoru...

RIEŠENIE

vykreslenie grafu

```
import numpy as np
%matplotlib notebook
import matplotlib.pyplot as plt
import ipywidgets as widgets
from maux import *

def f(X):
    return 2 * X ** 4 - 7 * X ** 3 + 5 * X ** 2 + 3 * X - 2
X = np.linspace(-0.70, 2.20, 29*100+1)

fig, ax = plt.subplots(figsize=(7,9))
fig.tight_layout()
init_subplot(ax)
ax.set_aspect('equal')
ax.set_title(r"$y = 2x^4-7x^3+5x^2+3x-2$")
ax.plot(X, f(X),c='black', linewidth=2)
```

spustenie editora

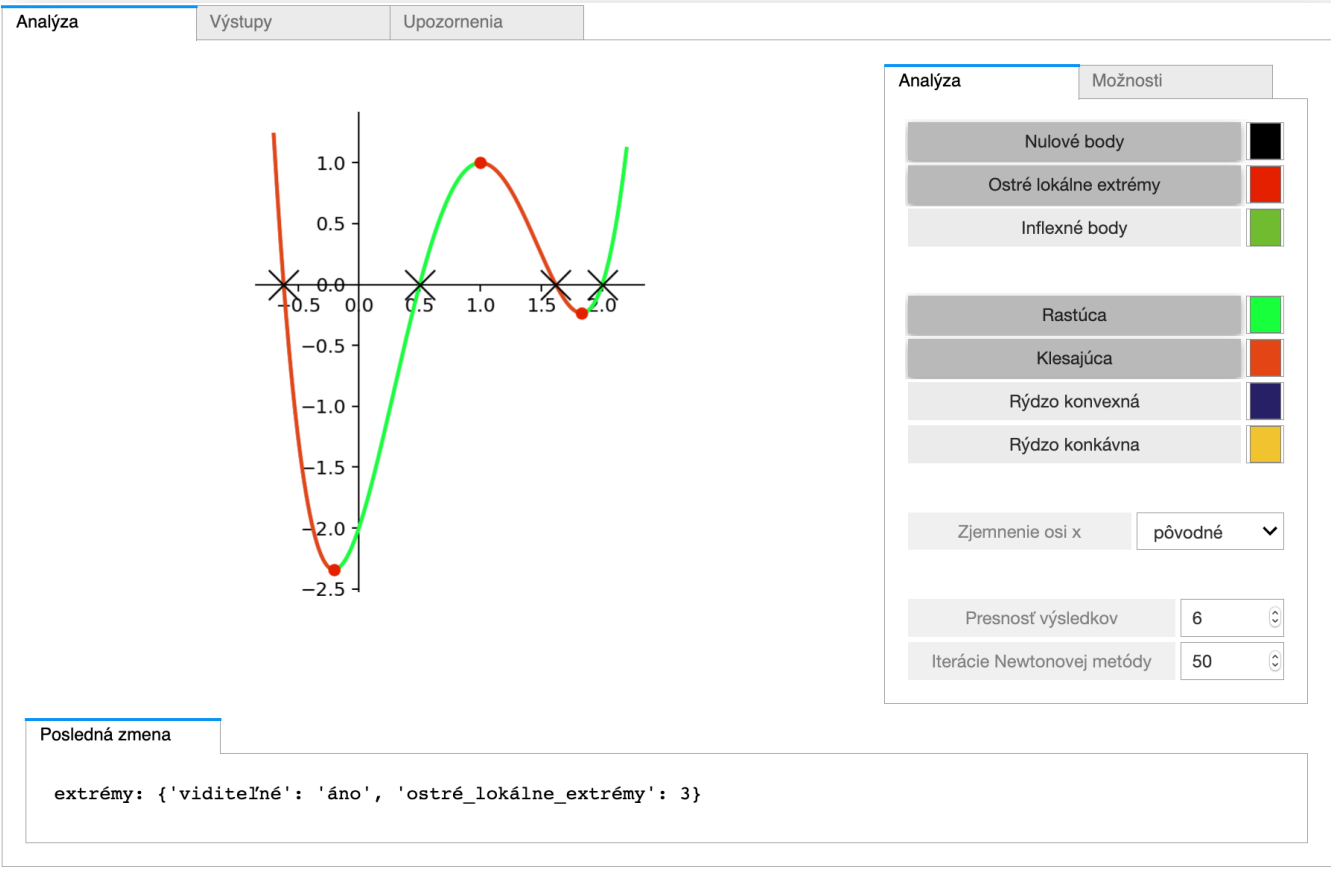
```
import numpy as np
%matplotlib notebook
import matplotlib.pyplot as plt
import ipywidgets as widgets
from maux import *
from jedit import editor

def f(X):
    return 2 * X ** 4 - 7 * X ** 3 + 5 * X ** 2 + 3 * X - 2
X = np.linspace(-0.70, 2.20, 29*100+1)

fig, ax = plt.subplots(figsize=(7,9))
fig.tight_layout()
init_subplot(ax)
ax.set_aspect('equal')
ax.set_title(r"$y = 2x^4-7x^3+5x^2+3x-2$")
ax.plot(X, f(X),c='black', linewidth=2)

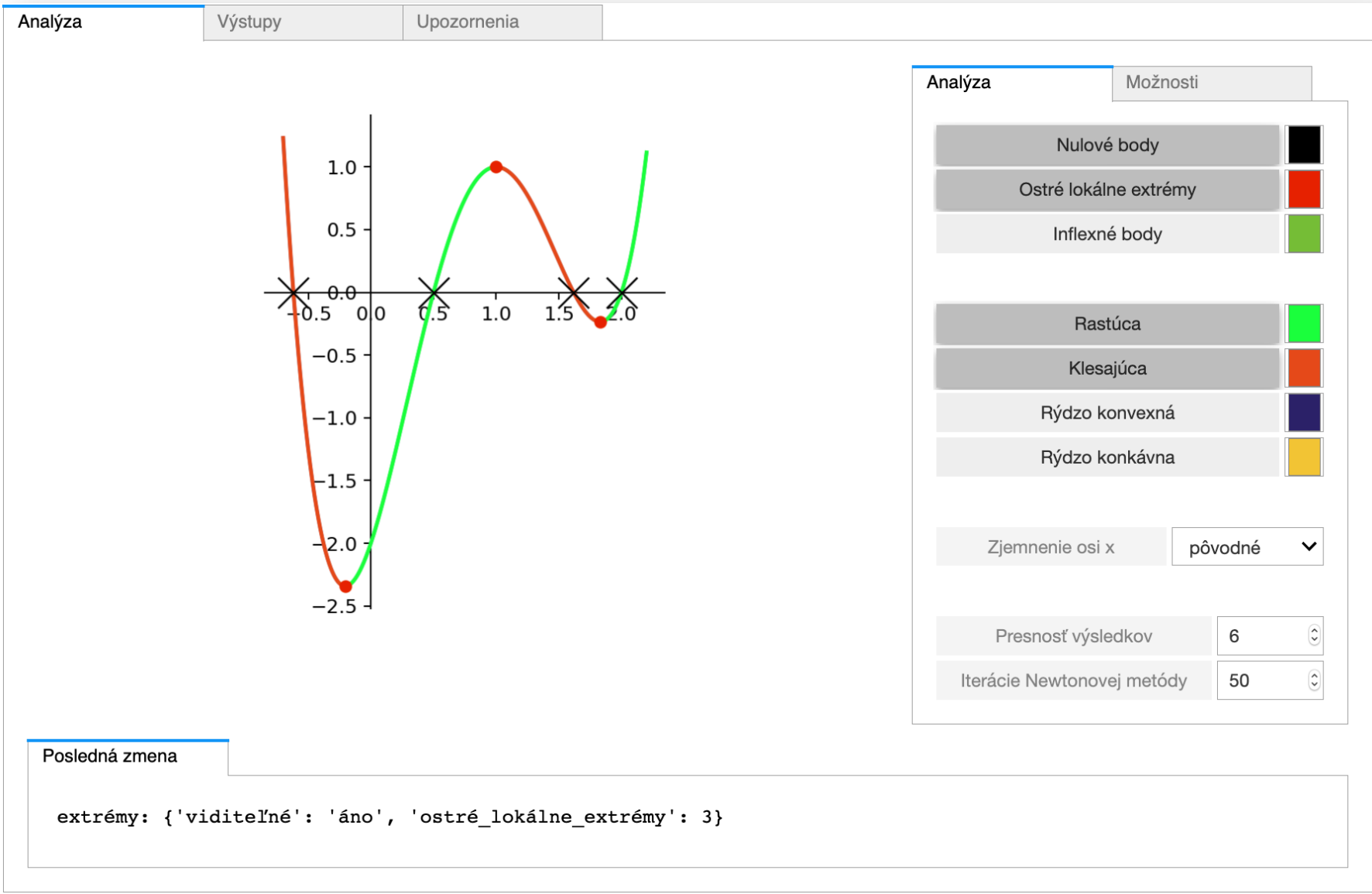
editor.figure=fig, axes=ax, function=f, intervals=[X])
```

analýza



cesta k editoru...

RIEŠENIE



Analýza Výstupy Upozornenia

```
[18.06.2020 18:40:50]
popis akcie: nulové body
derivácie: 0
metóda: Secant
maxiter: 50
počet: 4
v_bodoch: [
  -0.618034,
  0.5,
  1.61803,
  2.0
]
[18.06.2020 18:40:48]
popis akcie: extrém
viditeľné: áno
ostré_lokálne_extrémy_v_bodoch: [
  -0.205,
  1.0,
  1.83
]
ostré_lokálne_minimá_v_bodoch: [
  -0.205,
  1.83
]
ostré_lokálne_maximá_v_bodoch: [
  1.0
]
[18.06.2020 18:40:46]
popis akcie: klesajúca
viditeľné: áno
interval_x: [
  (-0.7, -0.206),
  (1.001, 1.829)
]
```

technologie v pozadí

programovací jazyk

Python 3.7.4

knižnice

Numpy

Scipy

matplotlib

ipywidgets

prostředí

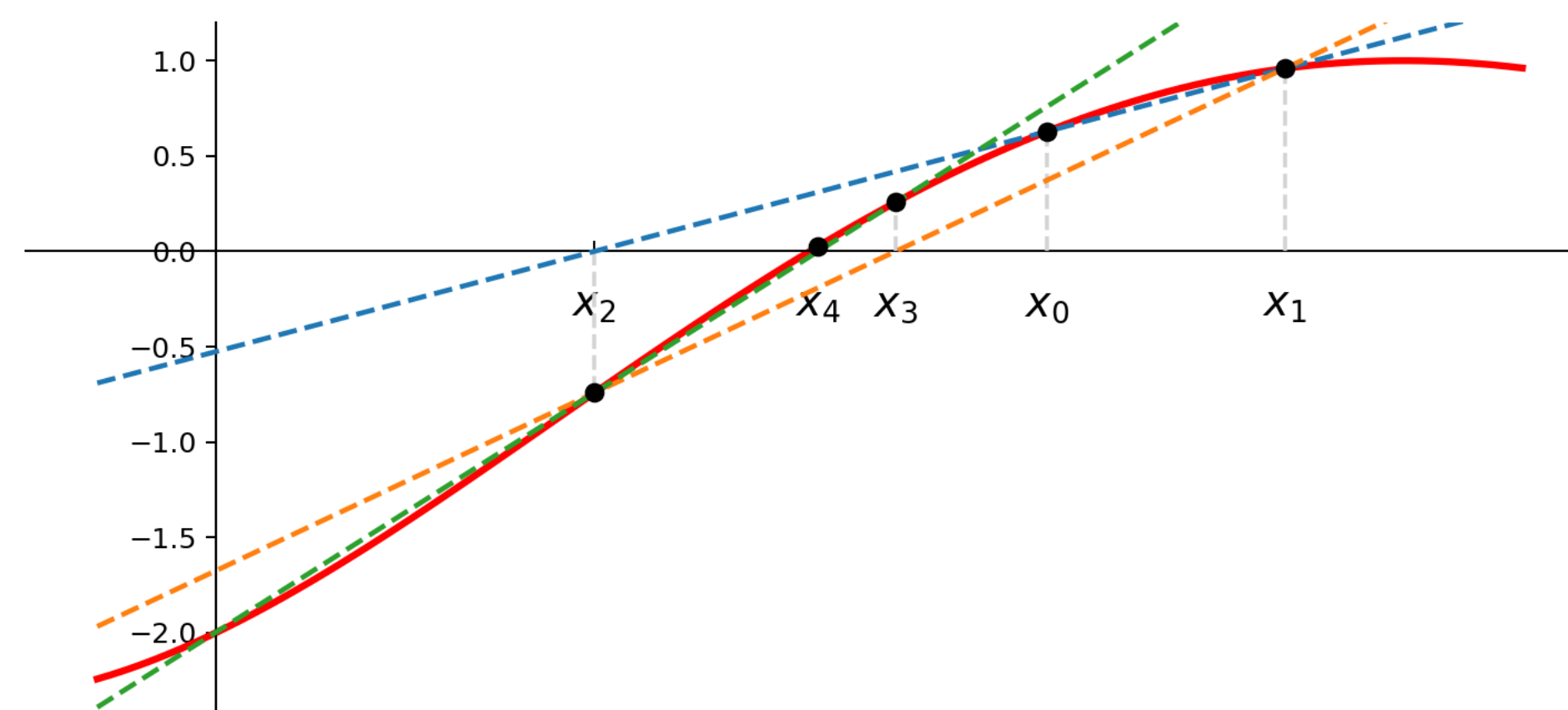
Anaconda

Jupyter Notebook

implementačné princípy

NUMERICKÝ VÝPOČET NULOVÝCH BODOV

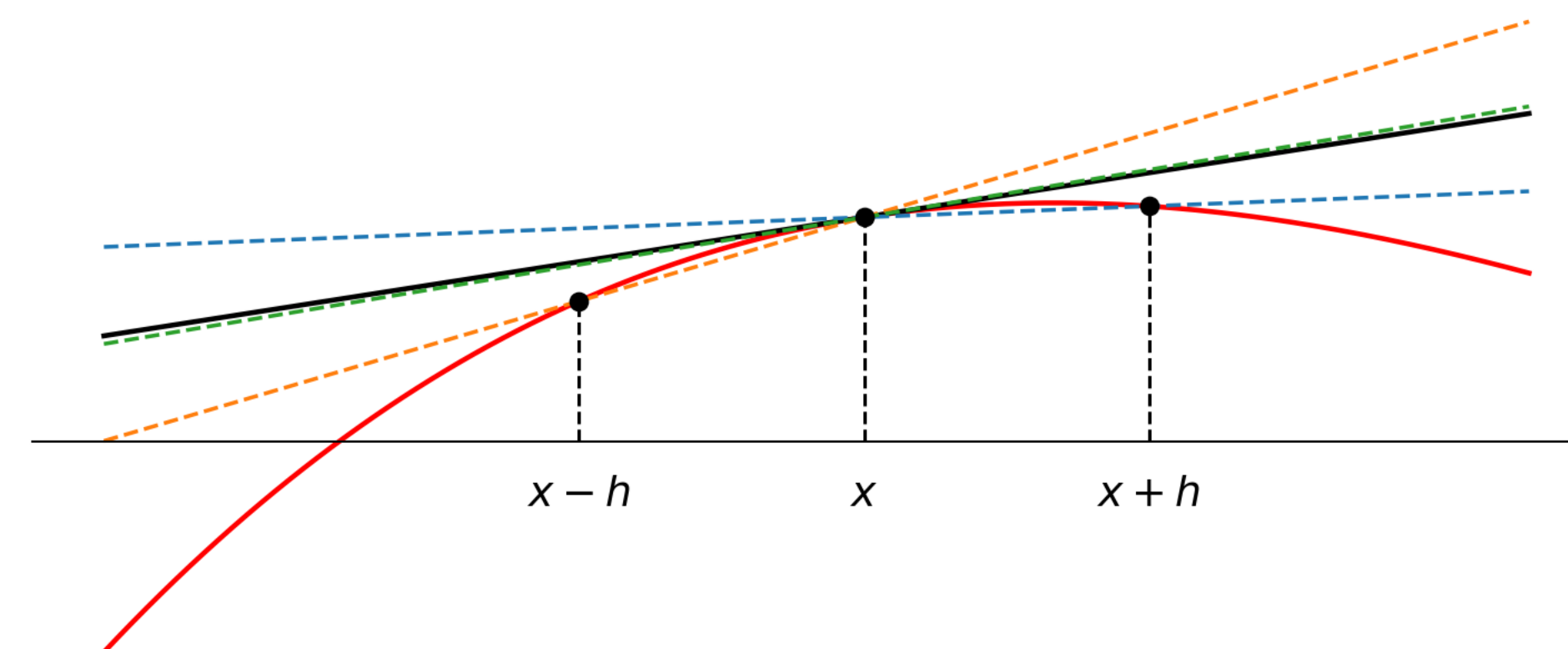
- Metóda sečníc
- Newtonova metóda
- `scipy.optimize.newton`



implementačné princípy

NUMERICKÝ VÝPOČET DERIVÁCIÍ

- študent nemusí vedieť derivovať
- editor však áno
- `scipy.misc.derivative` (centrálna diferencia)



implementačné princípy

CHARAKTERIZAČNÉ VETY O VYŠETROVANÍ PRIEBEHU FUNKCIE POMOCOU
DIFERENCIÁLNEHO POČTU

Veta 2. *Nech funkcia f je n -krát diferencovateľná vo vnútornom bode x množiny $D(f)$.
Nech ďalej $f'(x) = \dots = f^{(n-1)}(x) = 0$ a $f^{(n)}(x) \neq 0$ ($n \geq 2$).*

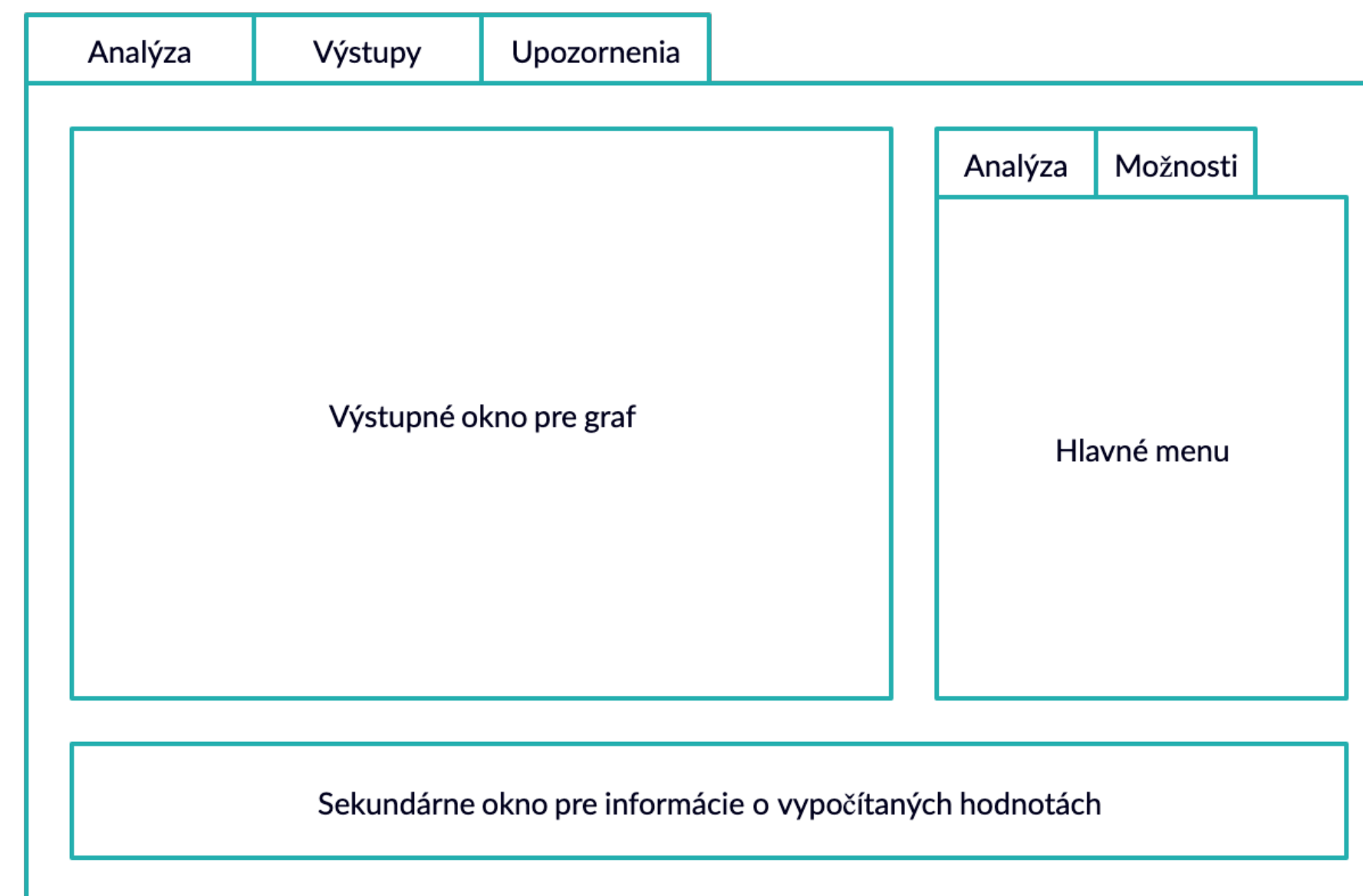
- Ak n je párne a $f^{(n)}(x) > 0$, tak funkcia f má v bode x ostré lokálne minimum.*
- Ak n je párne a $f^{(n)}(x) < 0$, tak funkcia f má v bode x ostré lokálne maximum.*
- Ak n je nepárne, tak funkcia f nemá v bode x lokálny extrém.*

```
def extremes(self, function):
    for i, interval in enumerate(function.intervals):
        primes1 = function.data['primes1'][f'interval{i}']
        for n in range(2, settings['extremes']['max_derivative'] + 1, 2):
            primes_n = function.data[f'primes{n}'].get(f'interval{i}', None)
            if primes_n is None:
                primes_n = add_derivative(function.func, interval, n)
            table = np.dstack((interval, primes1, primes_n))[0]
            candidates = table[table[:, 1] == 0]
            if len(candidates) == 0: break
            if numpy.any(candidates[candidates[:, 2] != 0]):
                minima = candidates[candidates[:, 2] > 0][:, 0]
                maxima = candidates[candidates[:, 2] < 0][:, 0]
                function.data['minima'][f'interval{i}'] = minima
                function.data['maxima'][f'interval{i}'] = maxima
                extrema = numpy.concatenate([minima, maxima])
                function.data['extremes'][f'interval{i}'] = extrema
            if not numpy.any(candidates[candidates[:, 2] == 0]): break
```

implementačné princípy

DIZAJN

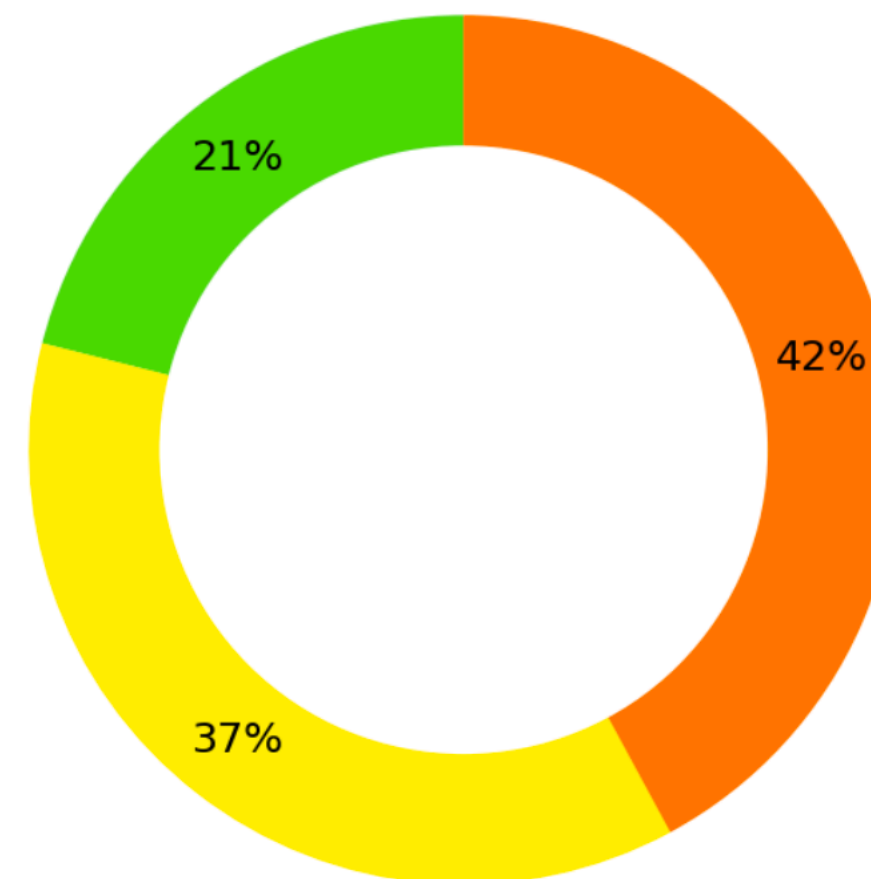
- intuitívny a prehľadný
- dôraz na sprievodné informácie
- dizajnové obmedzenia implementačného prostredia



prevádzkové výsledky

NASADENIE EDITORA DO VÝUČBY

- používaný na laboratóriach v LS
- testovaný študentami
- vývoj prispôsobený požiadavkám
- užitočnosť vyhodnotená dotazníkom



Aké je vaše celkové vnímanie použitia editora v súvislosti s preberanou látkou na laboratóriách?

- Bodaj by sa v ňom robilo všetko
- Som rád/a, keď ho môžem použiť
- Niekeddy mi ho napadne použiť
- Vôbec ho nepoužívam, je pre mňa zbytočný

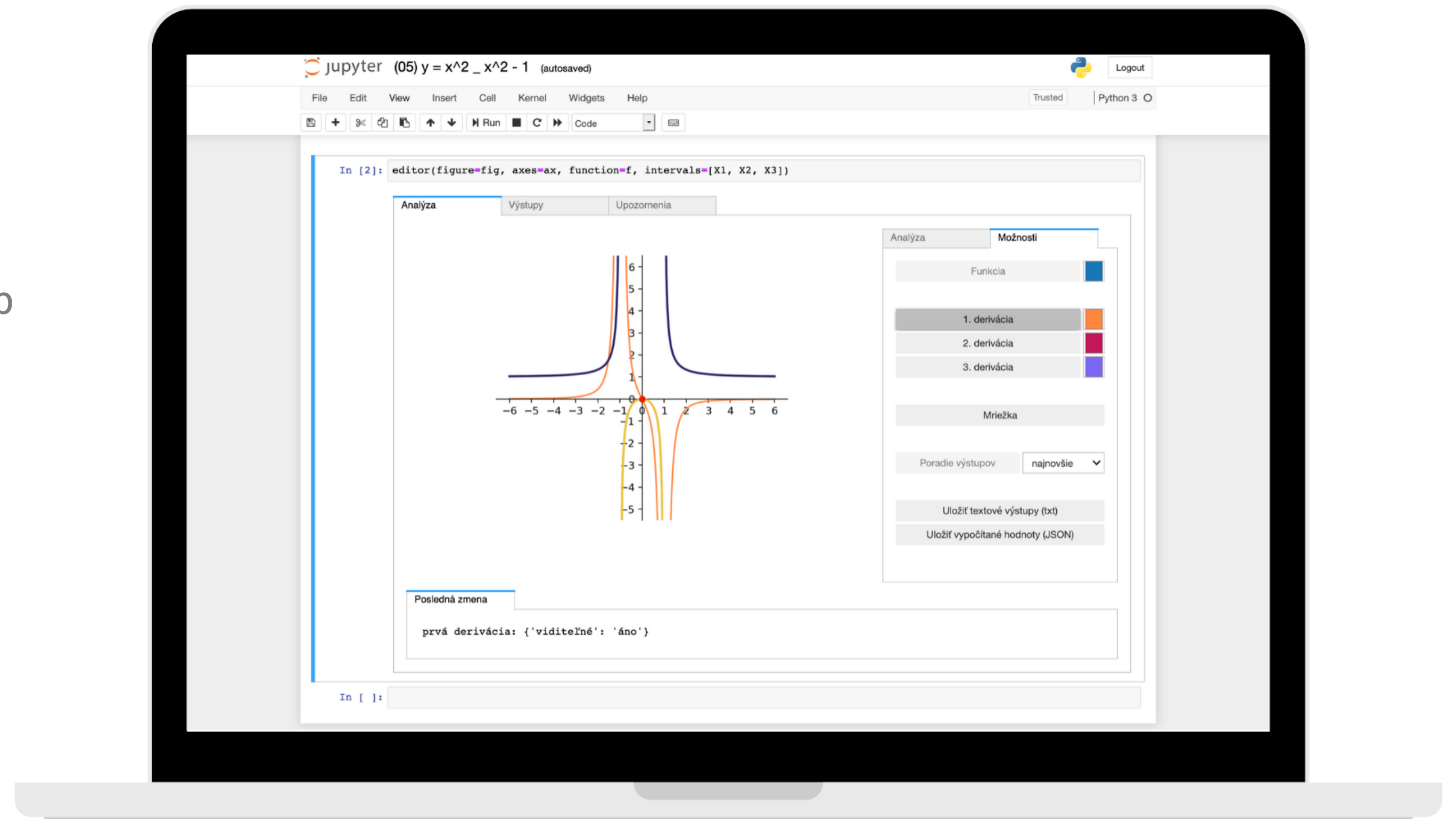
ukážka programu

VIDEO

zhrnutie a pohľad do budúcnosti

ZÁVER

- vývoj pokračuje
- pridanie ďalších funkcionalít pre čo najväčšiu elimináciu výpočtových chýb
- používanie aj v ďalších rokoch
- sprístupnenie pod GNU AGPL 3.0



d'akujem za pozornosť

Z Á V E R

otázky

ŠKOLITEL

otázky

OPONENTKA