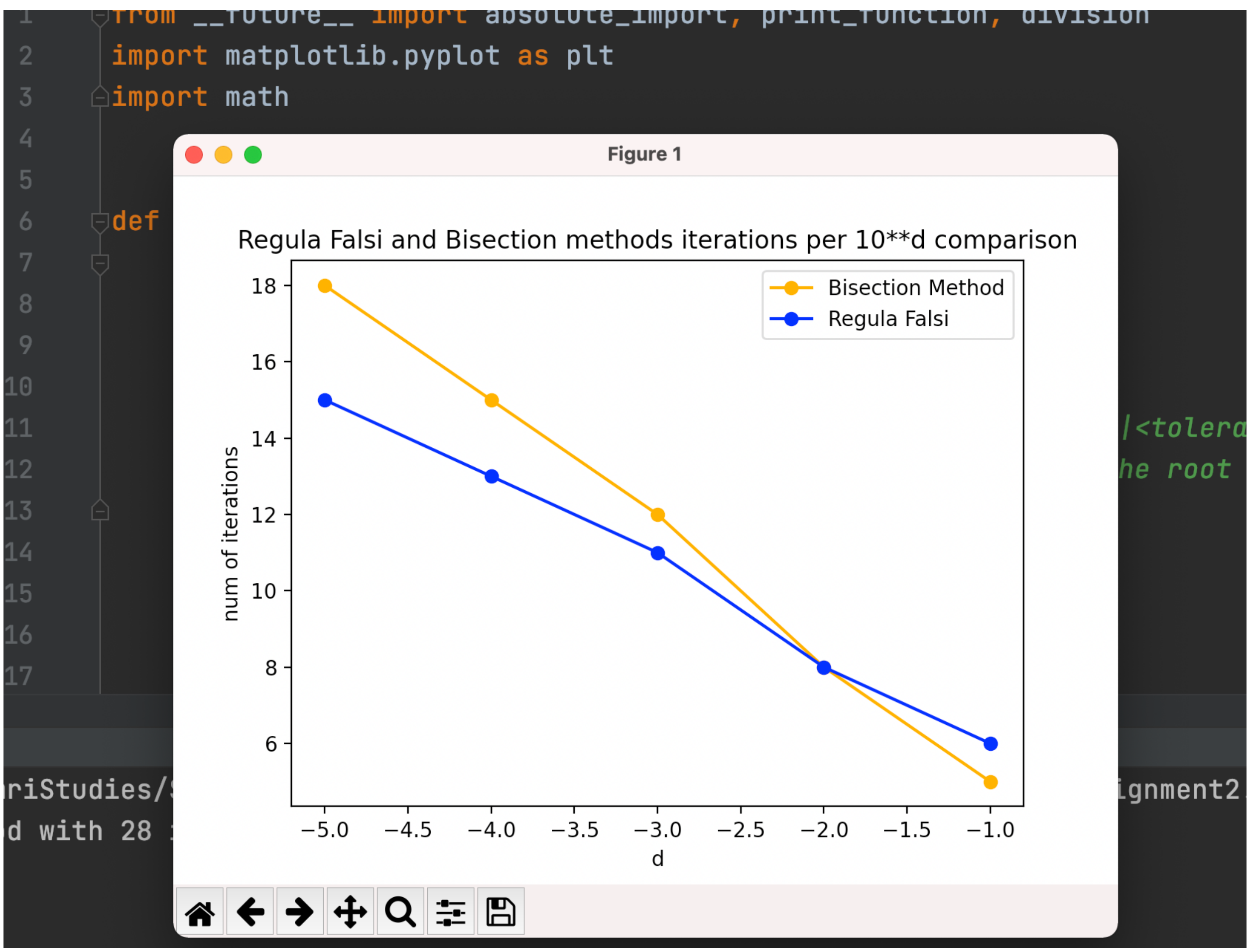
Introduction to Numerical Analysis Assignment 2

שאלה 1:

א. שיטת החצייה עם $tolerance = 10^{-8}$ יצא לנו 1.8349351603537798 לאחר 28 איטרציות.

ב. השוואה במספר האיטרציות כתלות בd כאשר מחפשים בשיטת החצייה ורגולר פולסי קירוב לתוצאה של סעיף א׳:

d	חצייה (קרוב של $10^{\wedge}d$ לשורש האמיתי)	רגולר פולסי (לקירוב של סעיף א׳)
-1	5	7
-2	8	9
-3	12	11
-4	15	14
-5	18	16



שאלה 2:

נגדיר $g(x)=\frac{x(x^2+3a)}{3x^2+a}$ כאשר $\sqrt{a}$ הוא שורש מכיוון ש:

$$g\left(\sqrt{a}\right)=\frac{\sqrt{a}(a+3a)}{3a+a}=\sqrt{a}$$

כעת נבדוק את סדר ההתכנסות ע״י מציאת R מינימלי כך ש- $g^R\left(\sqrt{a}\right)\neq 0$.

תחילה נסדר בצורה נוחה יותר את הביטוי:

$$g(x)=\frac{x(x^2+3a)}{3x^2+a}=\frac{x^3+3ax}{3x^2+a}$$

כעת נגזור פעם ראשונה:

$$g'(x)=\frac{\overbrace{(3x^2+3a)\cdot(3x^2+a)}^{9x^4+12ax^2+3a^2}-\overbrace{(x^3+3ax)\cdot 6x}^{6x^4+18ax^2}}{(3x^2+a)^2}=\frac{3x^4-6ax^2+3a^2}{(3x^2+a)^2}$$

נציב את $\sqrt{a}$:

$$g'\left(\sqrt{a}\right)=\frac{3a^2-6a^2+3a^2}{(3a+a)^2}=0$$

נגזור פעם שנייה:

$$\begin{aligned}g''(x)&=\frac{(12x^3-12ax)\cdot(3x^2+a)^2-(3x^4-6ax^2+3a^2)\cdot 12x(3x^2+a)}{(3x^2+a)^4}\\&=\frac{(3x^2+a)}{(3x^2+a)}\cdot\frac{(12x^3-12ax)\cdot(3x^2+a)-(3x^4-6ax^2+3a^2)\cdot 12x}{(3x^2+a)^3}\\&=\frac{\overbrace{(12x^3-12ax)\cdot(3x^2+a)}^{36x^5-24ax^3-12a^2x}-\overbrace{(3x^4-6ax^2+3a^2)\cdot 12x}^{36x^5-72ax^3+36a^2x}}{(3x^2+a)^3}=\frac{48x^3-48a^2x}{(3x^2+a)^3}\\&=\frac{48ax(x^2-a)}{(3x^2+a)^3}=\frac{48ax^3-48a^2x}{(3x^2+a)^3}\end{aligned}$$

נציב $\sqrt{a}$:

$$g''\left(\sqrt{a}\right)=\frac{48a\sqrt{a}\,\overbrace{(a-a)}^{0}}{(3a+a)^3}=0$$

נגזור פעם שלישית:

$$\begin{aligned}g^{(3)}(x)&=\frac{(144ax^2-96a^2)(3x^2+a)^{31}-(48ax^3-48a^2x)\cdot 18x(3x^2+a)^2}{(3x^2+a)^{64}}\\&=\frac{\overbrace{(144ax^2-96a^2)(3x^2+a)}^{432ax^4+\overbrace{144a^2x^2-288a^2x^2}^{-144a^2x^2}-96a^3}-\overbrace{(48ax^3-48a^2x)\cdot 18x}^{864ax^4-864a^2x^2}}{(3x^2+a)^4}\\&=\frac{-432ax^4+720a^2x^2-96a^3}{(3x^2+a)^4}\end{aligned}$$

נציב $\sqrt{a}$:

$$g^3(\sqrt{a}) = \frac{-432a \cdot a^2 + 720a^2 \cdot a - 96a^3}{(3a+a)^4} = \frac{192a^3}{256a^{41}} = \frac{3}{4a} \neq 0$$

ולכן סדר ההתכנסות הוא $R = 3$.

שאלה 3:

א. נתונה הפונקציה $f(x) = 2x^2 - 10$.
נכתוב את איטרציית ניוטון עבור פונקציה זו.
נחשב את הנגזרת כדי לבנות את הפונקצייה לאירטציית ניוטון:

$$f'(x) = 4x$$

כפי שלמדנו בכיתה איטרציית ניוטון היא:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

במקרה שלנו:

$$x_{n+1} = x_n - \frac{2x_n^2 - 10}{4x_n} = \frac{2x_n^2 - x_n^2 + 5}{2x_n} = \frac{x_n^2 + 5}{2x_n}$$

ב. ע״מ למצוא את תחום ההתכנסות נגדיר את פונקציית האיטרצייה לפי מה שמצאנו בסעיף א׳:

$$g(x) = \frac{x^2+5}{2x}$$

$$g'(x) = \frac{4x^2 - 2x^2 - 10}{4x^2} = \frac{x^2-5}{2x^2}$$

נבדוק את התנאים הדרושים:

1. פונקציית כיווץ: $|g'(x)| < 1$

$$\frac{x^2-5}{2x^2} < 1 \leftrightarrow x^2 - 5 < 2x^2 \leftrightarrow -5 < x\text{^}2 \leftrightarrow always\ true$$

$$-1 < \frac{x^2-5}{2x^2} \leftrightarrow -2x^2 < x^2 - 5 \leftrightarrow -3x^2 < -5 \leftrightarrow 5 < 3x^2 \leftrightarrow \frac{5}{3} < x^2 \leftrightarrow \sqrt{\frac{5}{3}} < |x|$$

כלומר כרגע התחום הוא $\sqrt{\frac{5}{3}} < x, x < -\sqrt{\frac{5}{3}}$.

2. כעת נבדוק כי לכל x בתחום, גם $g(x)$ בתחום.

יהי $\boldsymbol{x_0}$ $\sqrt{\frac{3}{5}} < \boldsymbol{x_0}$:

$$g(x_0) = \frac{x_0^2+5}{2x_0}$$

$$\sqrt{\frac{5}{3}} < g(x_0) = \frac{x_0^2+5}{2x_0} \leftrightarrow 0 < x_0^2 - \sqrt{\frac{5}{3}}x_0 + 5$$

נשים לב כי זה מתקיים תמיד מכיוון שאין ל-$g(x_0)$ אפשרות להיות 0 מכיוון שבדטרמיננטה של המשוואה הריבועית $\sqrt{\frac{5}{3}-4\cdot 5}<0$ אז אין פתרון למשוואה הריבועית עבור שוויון ל0, וגם היא רציפה וחיובית בנקודה $x_0=2$: $2^2-\sqrt{\frac{5}{3}}\cdot 2+5>0$ אזי הביטוי לכל x_0 בתחום.

יהי $\boldsymbol{x_0} > -\sqrt{\frac{3}{5}}$:

$$-\sqrt{\frac{\mathbf{5}}{\mathbf{3}}} > \boldsymbol{g(x_0)} = \frac{\boldsymbol{x_0^2+5}}{\boldsymbol{2x_0}} \overset{\text{הפיכת סימן בגלל במספר כפל שלילי}}{\Leftrightarrow} x_0^2+\sqrt{\frac{5}{3}}x_0+5>0$$

נשים לב כי זה מתקיים תמיד מכיוון שאין ל-$g(x_0)$ אפשרות להיות 0 מכיוון שבדטרמיננטה של המשוואה הריבועית $\sqrt{\frac{5}{3}-4\cdot 5}<0$ אז אין פתרון למשוואה הריבועית עבור שוויון ל0, וגם היא רציפה וחיובית בנקודה $x_0=-2$: $(-2)^2+\sqrt{\frac{5}{3}}\cdot(-2)+5>0$ אזי הביטוי לכל x_0 בתחום.

3. רציפות: קל לראות כי $g(x)$ רציפה.

סה״כ הראנו כי לכל $|x|>\sqrt{\frac{5}{3}}$ מתקיימים תנאי משפט 4 ולכן שיטת ניוטון תתכנס לשורש כנדרש.

שאלה 4:

א. תהי f פונ׳ גזירה וקמורה ממש ב-$\mathbb{R}$ ו- $f'(x)>0$ לכל x. נראה כי ל-f שורש יחיד.
נניח בשלילה כי קיימים שני שורשים: x_1,x_2. אזי $f(x_1)=0, f(x_2)=0$. נניח בה״כ $x_2>x_1$ ולכן גם $x_2-x_1>0$. הפונ׳ קמורה ממש ולכן מההגדרה:

$$f(x_2)=0>f(x_1)+f'(x_1)\cdot(x_2-x_1)=0+\overbrace{f'(x_1)}^{\text{מהנתון חיובי}}\cdot\overbrace{(x_2-x_1)}^{\text{גדול מ0}}>0\to 0>0$$

סתירה!

ב. יהי α שורש f ו-x_0 ניחוש התחלתי של איטרציית ניוטון רפסון.
אם $x_0=\alpha$ סיימנו.

אחרת, נראה כי סדרת הערכים $\{x_n\}$ של איטרציית ניוטון רפסון מונוטונית וחסומה, ולכן נסיק כי היא מתכנסת. לאחר מכן נראה כי ההתכנסות היא לשורש.
יהי x_n סדרת הנקודות המתקבלת ע״י איטרציית ניוטון.
נראה כי $x_{n+1}<x_n, \alpha<x_n$ ואז נסיק כי הסדרה היא מונוט

<u>חסימות</u>: נראה כי $\forall n\geq 1: \alpha<x_n$:

מהגדרת פונ׳ קמורה ממש:

$$f(b) > f(a) + f'(a)(b-a)$$

נציב $b=\alpha, a=x_n$

$$f(\alpha) > f(x_n) + f'(x_n)(\alpha - x_n)$$
$$0 > f(x_n) + f'(x_n)(\alpha - x_n)$$
$$-f(x_n) > f'(x_n)(\alpha - x_n)$$
$$-\frac{f(x_n)}{f'(x_n)} \overset{\text{חילוק במספר חיובי}}{>} \alpha - x_n$$
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} > \alpha$$

הראנו כי $x_{n+1} > \alpha$ לכל $n \geq 0$ לכן $x_n > \alpha$ לכל $n \geq 1$.

<u>מונוטוניות:</u> נראה כי $x_n > x_{n+1}$ לכל $n \geq 1$.
נסמן את האיבר הבא באיטרציית ניוטון:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

הראנו חסימות מלמטה ע״י $x_n > \alpha$ לכל $n \geq 1$ ו-f מונ׳ עולה לכן $f(x_n) > f(\alpha) = 0$.
כיוון שהנגזרת חיובית נקבל כי $\frac{f(x_n)}{f'(x_n)} > 0$.
לכן סה״כ:

$$x_{n+1} = x_n - \overbrace{\frac{f(x_n)}{f'(x_n)}}^{>0} < x_n - 0 = x_n$$

כנדרש.

<u>התכנסות לשורש:</u>
הראנו כי $x_n > x_{n+1}$ לכן סדרת הנקודות מונוטונית יורדת. כמו כן הראנו כי $x_n > \alpha$ לכל $n \geq 1$ לכן הסדרה חסומה מלמטה לכן מתכנסת. נסתכל על גבול הסדרה:

$$\lim_{n\to\infty} x_n = a$$

כעת נציב את הגבול בהגדת x_n:

$$a = \lim_{n\to\infty} x_n = \overbrace{a - \frac{f(a)}{f'(a)}}^{\text{הצבת הגבול באיבר הבא בסדרה}}$$
$$f(a) = 0$$

לכן הסדרה מתכנסת לשורש הפונקציה כנדרש.

ג. עבור $f'(x) < 0$ היינו מסיקים כי f מונוטונית יורדת.
- בהוכחת חסימות היינו מצליחים לחסום את x_n מלמעלה ע״י השורש (יהיה שינוי כיוון באי שוויון בעת החלוקה במספר שלילי במקום חיובי).
- בהוכחת מונוטוניות תשתנה המסקנה כי מחסירים את $\frac{f(x_n)}{f'(x_n)}$, אזי המכנה שלילי (כי הפונקציה מונוטונית יורדת), והמונה חיובי מכיוון שבחסימות נראה כי $x_n < \alpha$ וגם הפונ׳ f

היא מונוטונית יורדת אזי $0 = f(\alpha) < f(x_n) \rightarrow \alpha > x_n$ כלומר המונה חיובי, אזי בכל איטרציה מחסירים מ-x_n איבר שלילי כלומר מגדילים אותו. לכן יתקיים $x_{n+1} > x_n$.

- בהוכחת התכנסות לשורש אין שינוי.

ד. אם היה נתון כי f קעורה ממש בכל התחום אז השינוי היחיד היה בהוכחת החסימות, ושם היינו מוכיחים כי $x_n < \alpha$ לכל $n \geq 1$ מכיוון שהיה מתשנה הכיוון של האי שוויון שהתבססנו עליו.

<u>שאלה 1:</u>

```
from __future__ import absolute_import, print_function,
division
import matplotlib.pyplot as plt
import math


def bisection_method(f, a, b, tolerance):
```

```
    """
    :param f: function to determine the find the root of
it
    :param a: left limit of the domain
    :param b: right limit of the domain
    :param tolerance: the tolerance which should suffice
|z-a'|<tolerance where a' is the root and z is the
returned value
    :return: z s.t |z-a'|<tolerance, # of iterations to
find the root
    """
    i = 0
    fb = f(b)
    fa = f(a)
    if fa * fb > 0:
        raise ValueError("f(a)*f(b) should be negative
but is {0}".format(fa * fb))
    elif fa == 0:
        return a, i
    elif fb == 0:
        return b, i

    limit = 2 * tolerance
    while math.fabs(a - b) >= limit:
        i += 1
        z = (a + b) / 2
        fz = f(z)
        if fz == 0:
            return z, i
        fa = f(a)
        if fa * fz < 0:
            b = z
        else:
            a = z

    return (a + b) / 2, i


def regula_falsi(f, x0, x1, tolerance, goal):
    """
    :param f: function to determine the find the root of
it
```

```python
    :param x0: initial guess s.t f(x0)*f(x1) < 0
    :param x1: inititial guess s.t f(x0)*f(x1) < 0
    :param tolerance: the tolerance to be clos to the
root
    :param goal: the goal to return
    :return: x s.t |f(x)| < tolerance
    """
    fx0 = f(x0)
    fx1 = f(x1)
    if fx0 == 0:
        return x0, 0
    elif fx1 == 0:
        return x1, 0
    elif fx0 * fx1 > 0:
        raise ValueError("f(x0)*f(x1) should be negative
but is {0}".format(fx0 * fx1))
    i = 0
    xi_minus_1 = x1
    xi_minus_2 = x0
    xi = float('inf')
    while math.fabs(xi - goal) >= tolerance:
        i += 1
        fxi_minus_1 = f(xi_minus_1)
        delta_f = fxi_minus_1 - f(xi_minus_2)
        delta_xi = xi_minus_1 - xi_minus_2
        xi = xi_minus_1 - fxi_minus_1 * delta_xi /
delta_f
        fxi = f(xi)
        if fxi == 0:
            return xi, i
        if fxi * fxi_minus_1 < 0:
            xi_minus_2 = xi
        else:
            xi_minus_1 = xi

    return xi, i


def f(x):
    return x ** 2 - 0.2 * x - 3
```

```
def main():
    #start a
    x0 = -1
    x1 = 4
    tolerance = 10 ** -8
    bisection_result, bisection_iter_num =
bisection_method(f, x0, x1, tolerance)
    print("got result {0} in bisection method with {1}
iterations.".format(bisection_result,
bisection_iter_num))
    #finish a
    #start b
    tolerance_axis = []
    bisection_iterations_axis = []
    falsi_iterations_axis = []
    for d in range(1, 6):
        d = -1 * d
        tolerance_axis.append(d)
        tolerance = 10 ** d
        bisection, bisection_iter = bisection_method(f,
x0, x1, tolerance)
        bisection_iterations_axis.append(bisection_iter)
        falsi_result, falsi_iter = regula_falsi(f, x0,
x1, tolerance, bisection_result)
        falsi_iterations_axis.append(falsi_iter)
    plt.plot(tolerance_axis, bisection_iterations_axis,
color='orange', marker="o", label='Bisection Method')
    plt.plot(tolerance_axis, falsi_iterations_axis,
color='blue', marker="o", label='Regula Falsi')
    plt.xlabel("d")
    plt.ylabel("num of iterations")
    plt.title("Regula Falsi and Bisection methods
iterations per 10**d comparison")
    plt.legend()
    plt.show()
    #finish b


if __name__ == '__main__':
    main()
```