<u>אנליזה נומרית - עבודה 1</u>

אילנה פרבוי – 318271640, פן אייל - 208722058

<u>שאלה 1</u>

א. נחשב המרה מעשרוני לבינארי עבור 0.1 :

ערך שלם	X2	מספר
0	0.2	0.1
0	0.4	0.2
0	0.8	0.4
1	1.6	0.8
1	1.2	0.6
0	0.4	0.2

$$(0.1)_{10} = \left(\mathbf{0.0\overline{0011}}\right)_{\mathbf{2}}$$

עבור $0.\tilde{1}$ הייצוג הוא של 23 ספרות ולכן הייצוג יהיה :

$$(0.\tilde{1})_{10} = \left(\mathbf{0.}\underbrace{\mathbf{00011001100110011001100}}_{\mathbf{23}\text{ ספרות}}\right)_{\mathbf{2}}$$

ב. חישוב השגיאה המוחלטת :

$$\Delta\left(0.\tilde{1}\right)_{10} = \left(\left|0.1 - 0.\tilde{1}\right|\right)_{10} = \left(\left|0.0\overline{0011} - 0.00011001100110011001100\right|\right)_2$$

$$= \left(0.\underbrace{00000000000000000000000}_{23\text{ ספרות}}\overline{1100}\right)_2 = (\mathbf{0.1 * 2^{-20}})_{\mathbf{10}}$$

חישוב השגיאה היחסית :

$$\delta\left(0.\tilde{1}\right)_{10} = \left(\frac{\Delta\left(0.\tilde{1}\right)}{0.1}\right)_{10} = \left(\frac{0.1 * 2^{-20}}{0.1}\right)_{10} = (\mathbf{2^{-20}})_{\mathbf{10}}$$

ג. נחשב את מספר הספרות הבינאריות המשמעותיות בקירוב $\left(0.\tilde{1}\right)_{10}$.

נחפש מהו הd הגדול ביותר כך ש : $\delta\left(0.\tilde{1}\right)_{10} \leq 2^{1-d}$

$$\delta\left(0.\tilde{1}\right)_{10} = 2^{-20} \leq 2^{1-d} \leftrightarrow -20 \leq 1 - d \leftrightarrow d \leq 21$$

כלומר $\boldsymbol{d = 21}$.

ד.

$$n_1 = \frac{8_{hours}}{(0.1)_{seconds}} = \frac{(8 * 60 * 60)_{seconds}}{(0.1)_{seconds}} = \frac{(28800)_{seconds}}{(0.1)_{seconds}} = (288000)_{10}$$

$$n_2 = \frac{8_{hours} + 2_{seconds}}{(0.1)_{seconds}} = \frac{(8 * 60 * 60)_{seconds} + 2_{seconds}}{(0.1)_{seconds}} = \frac{(28802)_{seconds}}{(0.1)_{seconds}}$$
$$= (288020)_{10}$$

$$\tilde{t_1} = 0.\tilde{1} * n_1 = 0.\tilde{1} * 288000$$

$$\tilde{t_2} = 0.\tilde{1} * n_2 = 0.\tilde{1} * 288020$$

$$\tilde{t} = \tilde{t_2} - \tilde{t_1} = 0.\tilde{1} * 288020 - 0.\tilde{1} * 288000 = 0.\tilde{1} * (288020 - 288000) = 0.\tilde{1} * 20$$

$$t = t_2 - t_1 = 0.1 * 288020 - 0.1 * 288000 = 0.1 * (288020 - 288000) = 0.1 * 20$$

$$\Delta\tilde{t} = |\tilde{t} - t| = \left|0.\tilde{1} * 20 - 0.1 * 20\right| = \left|20 * \left(0.\tilde{1} - 0.1\right)\right| = 20 * \Delta\left(0.\tilde{1}\right)$$
$$= 20 * 0.1 * 2^{-20} = \mathbf{2^{-19}}$$

$$\delta\tilde{t} = \frac{\Delta\tilde{t}}{|t|} = \frac{2^{-19}}{0.1 * 20} = \mathbf{2^{-20}}$$

ה. מכיוון שהיחס $\tilde{t} = \tilde{t_2} - \tilde{t_1}$ נשמר, אנו נקבל את אותה התוצאה, כפי שניתן לראות:

$$n_1 = \frac{100_{hours}}{(0.1)_{seconds}} = \frac{(100 * 60 * 60)_{seconds}}{(0.1)_{seconds}} = \frac{(360000)_{seconds}}{(0.1)_{seconds}} = (3600000)_{10}$$
$$n_2 = \frac{100 + 2_{seconds}}{(0.1)_{seconds}} = \frac{(100 * 60 * 60)_{seconds} + 2_{seconds}}{(0.1)_{seconds}} = \frac{(360002)_{seconds}}{(0.1)_{seconds}}$$
$$= (3600020)_{10}$$

$$\tilde{t_1} = 0.\tilde{1} * n_1 = 0.\tilde{1} * 3600000$$

$$\tilde{t_2} = 0.\tilde{1} * n_2 = 0.\tilde{1} * 3600020$$

$$\tilde{t} = \tilde{t_2} - \tilde{t_1} = 0.\tilde{1} * 3600020 - 0.\tilde{1} * 3600000 = 0.\tilde{1} * (3600020 - 3600000) = 0.\tilde{1} * 20$$

$$t = t_2 - t_1 = 0.1 * 3600020 - 0.1 * 3600000 = 0.1 * (3600020 - 3600000) = 0.1 * 20$$

$$\Delta\tilde{t} = |\tilde{t} - t| = \left|0.\tilde{1} * 20 - 0.1 * 20\right| = \left|20 * \left(0.\tilde{1} - 0.1\right)\right| = 20 * \Delta\left(0.\tilde{1}\right)$$
$$= 20 * 0.1 * 2^{-20} = \mathbf{2^{-19}}$$

$$\delta\tilde{t} = \frac{\Delta\tilde{t}}{|t|} = \frac{2^{-19}}{0.1 * 20} = \mathbf{2^{-20}}$$

ו. נשים לב כי רק החישוב של $\tilde{t_2}$ והחישובים התלויים בו ישתנו:

$$n_1 = (288000)_{10}$$
$$n_2 = (288020)_{10}$$

$$\tilde{t_1} = 0.\tilde{1} * n_1 = 0.\tilde{1} * 288000$$

$$\tilde{t_2} = 0.1 * n_2 = 0.1 * 288020 = 28802$$

נשים לב:

$$(0.\tilde{1})_{10} = (0.1)_{10} - \Delta\left(0.\tilde{1}\right)_{10} = 0.1 - 0.1 * 2^{-20} = 0.1 * (1 - 2^{-20})$$

$$\tilde{t} = \tilde{t_2} - \tilde{t_1} = 28802 - 0.\tilde{1} * 288000 = 28802 - 0.1 * (1 - 2^{-20}) * 288000 = 2.02746582$$

$$t = t_2 - t_1 = 0.1 * 288020 - 0.1 * 288000 = 0.1 * (288020 - 288000) = 0.1 * 20 = 2$$

$$\Delta\tilde{t} = |\tilde{t} - t| = |2.02746582 - 2| = \mathbf{0.02746582}$$

$$\delta\tilde{t} = \frac{\Delta\tilde{t}}{|t|} = \frac{0.02746582}{2} = \mathbf{0.0137329102}$$

ז. בדומה לסעיף ה' נקבל כי n_1, n_2 יהיו שווים ל:

$$n_1 = (3600000)_{10}$$
$$n_2 = (3600020)_{10}$$

$$\tilde{t_1} = 0.\tilde{1} * n_1 = 0.\tilde{1} * 3600000$$

$$\tilde{t_2} = 0.1 * n_2 = 0.1 * 3600020 = 360002$$

$$\tilde{t} = \tilde{t_2} - \tilde{t_1} = 360002 - 0.\tilde{1} * 3600000 = 360002 - 0.1 * (1 - 2^{-20}) * 3600000 = 2.343322754$$

$$t = t_2 - t_1 = 0.1 * 3600020 - 0.1 * 3600000 = 0.1 * (3600020 - 3600000) = 0.1 * 20 = 2$$

$$\Delta\tilde{t} = |\tilde{t} - t| = |2.343322754 - 2| = \mathbf{0.343322754}$$

$$\delta\tilde{t} = \frac{\Delta\tilde{t}}{|t|} = \frac{0.343322754}{2} = \mathbf{0.171661377}$$

<u>שאלה 2</u>

א. <u>עבור *single* בייצוג 32 ביט:</u>

מנטיסה	חזקה	סימן	Bias	Normal\Un
$\underbrace{0 \ldots 0}_{\text{23 ביטים}}$	$\underbrace{0 \ldots 0}_{\text{7 ביטים}}1$	0	$2^7 - 1 = 127$	*Normal*

זהו המספר החיובי הקטן ביותר הניתן להציג בדיוק Single Normal, כלומר:

$$x = +1 * \left(1.\underbrace{0 \ldots 0}_{\text{23 ביטים}}\right)_2 * 2^{-126} = +1 * 1 * 2^{-126} = \mathbf{2^{-126}}$$

מנטיסה	חזקה	סימן	Bias	Normal\Un
$\underbrace{0 \ldots 0}_{\text{22 ביטים}}\ 1$	$\underbrace{0 \ldots 0}_{\text{7 ביטים}}1$	0	$2^7 - 1 = 127$	*Non-Normal*

זהו המספר החיובי הקטן ביותר הניתן להציג בדיוק Single Non-Normal, כלומר:

$$x = +1 * \left(0.\underbrace{0 \ldots 0}_{\text{22 ביטים}}\ 1\right)_2 * 2^{-126} = +1 * 2^{-23} * 2^{-126} = +1 * 2^{-149} = \mathbf{2^{-149}}$$

ב. עבור *double* בייצוג 64 ביט:

Normal\Un	Bias	סימן	חזקה	מנטיסה
Normal	$2^{10}-1=1023$	0	$\underbrace{0\ldots0}_{\text{10 ביטים}}\ 1$	$\underbrace{0\ldots0}_{\text{52 ביטים}}$

זהו המספר החיובי הקטן ביותר הניתן להציג בדיוק Double Normal, כלומר:

$$x=+1*\left(1.\underbrace{0\ldots0}_{\text{52 ביטים}}\right)_2*2^{-1022}=+1*1*2^{-1022}=\mathbf{2^{-1022}}$$

Normal\Un	Bias	סימן	חזקה	מנטיסה
Non-Normal	$2^{10}-1=1023$	0	$\underbrace{0\ldots0}_{\text{10 ביטים}}\ 1$	$\underbrace{0\ldots0}_{\text{51 ביטים}}\ 1$

זהו המספר החיובי הקטן ביותר הניתן להציג בדיוק Double Non-Normal, כלומר:

$$x=+1*\left(0.\underbrace{0\ldots0}_{\text{51 ביטים}}\ 1\right)_2*2^{-1022}=+1*2^{-52}*2^{-1022}=+1*2^{-1074}=\mathbf{2^{-1074}}$$

שאלה 3

א.

$$\Delta(\tilde{x}-\tilde{y})=|(x-y)-(\tilde{x}-\tilde{y})|=|x-\tilde{x}+y-\tilde{y}|$$

נקבל מאי שוויון המשולש כי:

$$\le|x-\tilde{x}|+|y-\tilde{y}|=\Delta\tilde{x}-\Delta\tilde{y}\ \blacksquare$$

ב. נראה תחילה כי $\delta(\tilde{x}*\tilde{y})\lesssim\delta\tilde{x}+\delta\tilde{y}$

נסמן: $\tilde{y}=y-\Delta y\,,\tilde{x}=x-\Delta x$

$$\Delta(\tilde{x}*\tilde{y})=|\tilde{x}*\tilde{y}-x*y|=|(x-\Delta x)*(y-\Delta y)-x*y|$$
$$=|x*y-x\Delta y-y\Delta x+\Delta x\Delta y-x*y|=|-x\Delta y-y\Delta x+\Delta x\Delta y|$$

$$\delta(\tilde{x}*\tilde{y})=\frac{\Delta(\tilde{x}*\tilde{y})}{|x*y|}=\frac{|-x\Delta y-y\Delta x+\Delta x\Delta y|}{|x*y|}$$

מאי שוויון המשולש נקבל:

$$\le\frac{|x\Delta y|}{|x*y|}+\frac{|y\Delta x|}{|x*y|}+\frac{|\Delta x\Delta y|}{|x*y|}=\frac{|\Delta y|}{|y|}+\frac{|\Delta x|}{|x|}+\frac{|\Delta x|}{|x|}*\frac{|\Delta y|}{|y|}=\delta\tilde{x}+\delta\tilde{y}+\delta(\tilde{x}*\tilde{y})$$

נשים לב שכאשר $\delta\tilde{x},\delta\tilde{y}$ קטנים אז $\delta(\tilde{x}*\tilde{y})$ כמעט מתאפס ולכן: $\delta(\tilde{x}*\tilde{y})\lesssim\delta\tilde{x}+\delta\tilde{y}$.

כעת נראה כי $\delta\left(\frac{\tilde{x}^2}{\tilde{y}^2}\right)\lesssim2(\delta\tilde{x}+\delta\tilde{y})$:

נשים לב: $\delta\left(\frac{\tilde{x}^2}{\tilde{y}^2}\right) = \delta\left(\tilde{x}^2 * \frac{1}{\tilde{y}^2}\right)$ ומהטענה הקודמת נקבל: $\delta\left(\tilde{x}^2 * \frac{1}{\tilde{y}^2}\right) \lesssim \delta\tilde{x}^2 + \delta\frac{1}{\tilde{y}^2}$

מכיוון ש: $\delta(\tilde{x}^2) = \delta(\tilde{x} * \tilde{x}) \lesssim \delta\tilde{x} + \delta\tilde{x}$ ומכיוון ש: $\delta\left(\frac{1}{\tilde{y}^2}\right) = \delta\left(\frac{1}{\tilde{y}} * \frac{1}{\tilde{y}}\right) \lesssim \delta\frac{1}{\tilde{y}} + \delta\frac{1}{\tilde{y}}$

נקבל $\delta\left(\tilde{x}^2 * \frac{1}{\tilde{y}^2}\right) \lesssim \delta\tilde{x} + \delta\tilde{x} + \delta\frac{1}{\tilde{y}} + \delta\frac{1}{\tilde{y}}$.

כעת נראה כי $\delta\left(\frac{1}{\tilde{y}}\right) = \delta(\tilde{y})$.

לפי הנוסחה:

$$\delta\left(\frac{1}{\tilde{y}}\right) = \frac{\left|\frac{1}{\tilde{y}} - \frac{1}{y}\right|}{\left|\frac{1}{y}\right|} = \frac{\left|\frac{y-\tilde{y}}{\tilde{y} * y}\right|}{\left|\frac{1}{y}\right|} = \left|\frac{y-\tilde{y}}{\tilde{y}}\right| = \left|\frac{\pm\Delta y}{y \mp \Delta y}\right| = \left|\frac{\pm\Delta y}{y \mp \Delta y} * \frac{y \pm \Delta y}{y \pm \Delta y}\right|$$
$$= \left|\frac{\pm\Delta y * y \pm \Delta y^2}{y^2 - \Delta y^2}\right|$$

בהנחה כי השגיאה המוחלטת קטנה, Δy^2 אפסית ולכן:

$$\delta\left(\frac{1}{\tilde{y}}\right) \approx \left|\frac{\pm\Delta y * y}{y^2}\right| = \left|\frac{\pm\Delta y}{y}\right| = \delta(\tilde{y})$$

לבסוף, נקבל כי:

$$\delta\left(\tilde{x}^2 * \frac{1}{\tilde{y}^2}\right) \lesssim \delta\tilde{x} + \delta\tilde{x} + \delta\tilde{y} + \delta\tilde{y} = 2(\delta\tilde{x} + \delta\tilde{y}) \blacksquare$$

<u>שאלה 4</u>

א. נפתח את השגיאה המוחלטת ע"פ כלל הגרדיאנט:

$$Z = Z(x,y) = e^{\alpha(x-y)}$$

$$\Delta\tilde{Z} \approx |\nabla Z(x,y)| * (\Delta\tilde{x}, \Delta\tilde{y}) = \left|\begin{pmatrix} \alpha e^{\alpha(x-y)} \\ -\alpha e^{\alpha(x-y)} \end{pmatrix}\right| * (\Delta x, \Delta y) = \left|\begin{pmatrix} \alpha Z \\ -\alpha Z \end{pmatrix}\right| * (\Delta\tilde{x}, \Delta\tilde{y})$$
$$= |\alpha Z| * \Delta\tilde{x} + |\alpha Z| * \Delta\tilde{y} = |\boldsymbol{\alpha Z}| * (\Delta\tilde{\boldsymbol{x}} + \Delta\tilde{\boldsymbol{y}})$$

$$\delta\tilde{Z} = \frac{\Delta\tilde{Z}}{|Z|} = \frac{|\alpha Z| * (\Delta\tilde{x} + \Delta\tilde{y})}{|Z|} = |\boldsymbol{\alpha}| * (\Delta\tilde{\boldsymbol{x}} + \Delta\tilde{\boldsymbol{y}})$$

ב.

$$\Delta\tilde{x} = \Delta\tilde{y} = 2$$
$$\rightarrow \delta\tilde{Z} = |\alpha| * (\Delta\tilde{x} + \Delta\tilde{y}) = 4 * |\alpha|$$

נרצה לבדוק מתי $\delta\tilde{Z} < 5\%$:

$$4 * |\alpha| < 0.05$$
$$\leftrightarrow |\alpha| < 0.0125$$
$$\leftrightarrow \boldsymbol{-0.0125 < \alpha < 0.0125}$$

שאלה 5

הקוד פייתון מצורף כקובץ נפרד, נדרש רק להריץ את הפונקציה *main*.

בנוסף מצורפת תמונה של הקוד בעמוד הבא.

א. לאחר הרצת הקוד נקבל:

```
approx for 70 itr: 0.281
real for 70 itr:   0.281
absolute error at 70 itr = 0.0
approx for 7000 itr: 1.0
real for 7000 itr:   28.001
absolute error at 7000 itr = 27.001
```

שגיאת הצובר גדולה יותר ככל שעובר הזמן מכיוון שלאחר וסכום הצובר מגיע ל-1 כל פעולת הוספה של מספר הקטן מ: 0.01 לו תגרור שהתוצאה תהיה גם כן 1 ללא שינוי.

בדוגמא שלנו מכיוון ש: $1 = 1 * 10^0$ וש: $0.004 = 4 * 10^{-3}$ שנחבר את המספרים, נקבל שההפרש בחזקות הוא מגודל 4, ולכן בעת החישוב של חיבור המנטיסות: $floor\left(1000 + \frac{4}{10}\right)$ יהיה שווה ל100 עם חזקה 3− ועל כן נקבל שהתוצאה לאחר החיבור עדיין שווה ל-$1 = 100 * 10^{-3}$

ב. לאחר הרצת הקוד נקבל:

```
the error difference between 70 itr and 72 itr = 0.0
the error difference between 8000 itr and 8002 itr = 0.008
```

שני ההפרשים שונים זה מזה בעקבות ההסבר מסעיף א'.

מכיוון ובאיטרציות 70 ו- 72 עדיין לא התחילו להיווצר שגיאות אנו נקבל כי אין שגיאות ביניהן.

אך לאחר 250 איטרציות $(250 * 0.004 = 1)$ בעקבות ההסבר בסעיף א', נקבל כי יתחילו להופיע שגיאות, וכפי המצופה השגיאה של האיטרציה ה- 8002 תהיה גדולה בפעמיים ערך החיבור המקורי במנטיסה מאשר האיטרציה ה- 8000 (מכיוון שהיו 2 יותר חיבורי מנטיסה בגודל 0.004 שלא נלקחו בחשבון)

```python
import math


def most_significant(num: int, digits_to_keep: int):
    """ return the digits_to_keep most significant digits of num
        return: (the kept digits, the number of non-significant (zeroedout) digits) """

    assert digits_to_keep > 0, 'digits_to_keep should be positive'
    num_digits = math.floor(math.log10(num)) + 1
    non_significant = max(0, num_digits - digits_to_keep)
    return (num // (10 ** non_significant)), non_significant


def special_adder(small_mantissa: int, small_exp: int, large_mantissa: int, large_exp: int):
    if small_exp > large_exp:
        # calculate c + acc
        # better to add larger (= acc) to smaller (= c)
        special_adder(large_mantissa, large_exp, small_mantissa, small_exp)

    # bring to the same exp
    diff_exp = large_exp - small_exp
    large_mantissa *= (10 ** diff_exp)
    new_mantissa, new_exp = most_significant(small_mantissa + large_mantissa, 3)
    new_exp += small_exp
    return new_mantissa, new_exp


def accumulator(n: int):
    acc_mantissa = 1
    acc_exp = -3
    c_mantissa = 4
    c_exp = -3
    for i in range(n):
        acc_mantissa, acc_exp = special_adder(acc_mantissa, acc_exp, c_mantissa, c_exp)

    return acc_mantissa * (10 ** acc_exp)


def print_num_and_error(n: int):
    acc = accumulator(n)
    print(f"approx for {n} itr: {acc}")
    real = round(0.001 + 0.004 * n, 5)
    print(f"real for {n} itr:   {real}")
    err = round(abs(real - acc), 5)
    print(f"absolute error at {n} itr = {err}")
    print("")
    return err


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    print("------------a------------")
    err_70 = print_num_and_error(70)
    err_7000 = print_num_and_error(7000)

    print("------------b------------")
    err_72 = print_num_and_error(72)
    err_8000 = print_num_and_error(8000)
    err_8002 = print_num_and_error(8002)

    diff_70_72 = round(abs(err_70 - err_72), 5)
    print(f"the error difference between 70 itr and 72 itr = {diff_70_72}")
    diff_8000_8002 = round(abs(err_8000 - err_8002), 5)
    print(f"the error difference between 8000 itr and 8002 itr = {diff_8000_8002}")
```