**#Mid point**

import sympy as sm

x = sm.symbols('x')

f\_x = x\*\*3+1

a\_limit = -2

b\_limit = 2

num = 8

h = (b\_limit-a\_limit)/num

interval\_points = []

for i in range(a\_limit\*10, b\_limit\*10+1, 5):

    interval\_points.append(i/10)

print("Point in interval with h: ", interval\_points)

mid\_point\_interval = []

for i in range(len(interval\_points)-1):

    mid\_point\_interval.append((interval\_points[i]+interval\_points[i+1])/2)

print("Mid\_points in interval:   ", mid\_point\_interval)

solution = 0

for i in mid\_point\_interval:

    solution += h\*f\_x.subs(x, i)

print("Solution by mid point rule: ", solution)

direct\_solution = sm.integrate(eval(str(f\_x)), (x, a\_limit, b\_limit))

print("Direct solution by integration: ", direct\_solution)

Error = abs(direct\_solution-solution)

print("error:   ", Error)

diff\_f = sm.diff(f\_x, x)

diff\_ff = sm.diff(diff\_f, x)

theory\_error = (b\_limit-a\_limit)\*diff\_ff.subs(x, b\_limit)/24\*h\*\*2

print("Theory error: ", theory\_error)

**OUTPUT:**

Изображение выглядит как текст

Автоматически созданное описание

**#Simpson’s rule**

import sympy as sm

import math

x = sm.symbols('x')

f\_x = x\*\*3+1

a\_limit = -2

b\_limit = 2

num = 8

h = (b\_limit-a\_limit)/num

interval\_points = []

for i in range(a\_limit\*10, b\_limit\*10+1, 5):

    interval\_points.append(i/10)

print("Point in interval with h: ", interval\_points)

solution = 0

for i in range(len(interval\_points)):

    if i==0 or i==len(interval\_points)-1:

        solution+=h/3\*f\_x.subs(x, interval\_points[i])

    else:

        if i%2==1:

            solution+=h/3\*f\_x.subs(x, interval\_points[i])\*4

        else:

            solution+=h/3\*f\_x.subs(x, interval\_points[i])\*2

print("Solution by Simpson rule: ", solution)

direct\_solution = sm.integrate(eval(str(f\_x)), (x, a\_limit, b\_limit))

print("Direct solution by integration: ", direct\_solution)

Error = abs(math.ceil(solution)- direct\_solution)

print("error:   ", Error)

diff\_f = sm.diff(f\_x, x)

diff\_ff = sm.diff(diff\_f, x)

diff\_fff = sm.diff(diff\_ff, x)

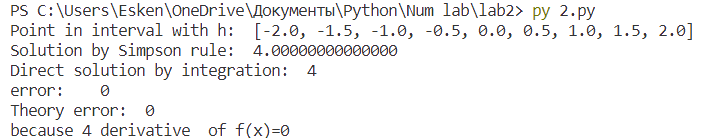
diff\_ffff = sm.diff(diff\_fff, x)

theory\_error = -1\*(b\_limit-a\_limit)\*diff\_ffff.subs(x, b\_limit)/180\*h\*\*4

print("Theory error: ", theory\_error)

print("because 4 derivative \_of\_f(x)=0")

**OUTPUT**



#Euler and Huens

import math

import sympy as sm

import matplotlib.pyplot as plt

import numpy as np

x = sm.symbols('x')

y = sm.symbols('y')

d\_y = 2\*y-3\*x

a\_limit = 4

b\_limit = 6

h = 0.5

y\_exact = 6/4\*x+3/4-27/4\*math.e\*\*(2\*x-8) #by diff methods

y\_arr = []

def dif\_y(x, y):

    return 2\*y -3\*x

x\_arr = list(np.arange(a\_limit, b\_limit+h, h))

for i in x\_arr:

    y\_arr.append(y\_exact.subs(x, i))

print(y\_arr)

def Euler():

    print('Euler:')

    y\_arr = [0]

    y = 0

    cnt = 0

    for i in x\_arr:

        y = y + h\*(dif\_y(i, y\_arr[-1]))

        y\_arr.append(y)

        print('X'+str(cnt)+':', i, "                 ", 'Y'+str(cnt+1)+':', y)

        cnt+=1

    return y\_arr[:-1]

def Huens():

    print('Huens:')

    y\_arr = [0]

    p\_arr = []

    cnt = 1

    for i in x\_arr:

        p = y\_arr[-1] + h\*(dif\_y(i, y\_arr[-1]))

        p\_arr.append(p)

        y = y\_arr[-1] + h/2\*(dif\_y(i, y\_arr[-1])+dif\_y(i+h, p))

        y\_arr.append(y)

        print('X'+str(cnt-1)+':', i, "            ", 'P'+str(cnt)+':', p, "       ", 'Y'+str(cnt)+':', y)

        cnt+=1

    return y\_arr[:-1]

Euler\_y\_arr = Euler()

Huens\_y\_arr = Huens()

error\_E = abs(y\_arr[-1]-Euler\_y\_arr[-1])

error\_H = abs(y\_arr[-1]-Huens\_y\_arr[-1])

print('Errors:  ')

print('Euler Error:         ', error\_E)

print('Huens Error:         ', error\_H)

plt.plot(x\_arr, y\_arr, color = "red", label = "Exact Graph")

plt.plot(x\_arr, Euler\_y\_arr, color = "green", label = "The Euler graph")

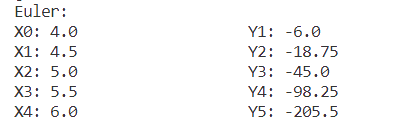
plt.plot(x\_arr, Huens\_y\_arr, color = "blue", label = "The Heun graph")

plt.xlabel("x")

plt.ylabel("y")

plt.legend()

plt.show()

Изображение выглядит как текст

Автоматически созданное описание

