# Bahria University,

## Karachi Campus



LAB EXPERIMENT NO.

**\_11\_**

LIST OF TASKS

|  |  |
| --- | --- |
| **TASK NO** | **OBJECTIVE** |
| 01 | Write a Python program that implements range – kutta methods of the orders mentioned against each part to approximate the 1 st order derivative of the following:  a. approximate y(0.2) for 𝒚′ = 𝒙−𝒚 , having x0 = 0, y0 = 1, and h = 0.1. (2nd order)  b. approximate y(0.3) for 𝒚′ = 𝟏 − 𝒚𝟑, having x0 = 0, y0 = 0, and h = 0.1. (3rd order)  c. approximate y(0.1) for 𝒚′ = −𝟐𝒚 + 𝒙𝟑𝒆−𝟐𝒙, having x0 = 0, y0 = 1, & h = 0.1. (4th order) |
|  |  |
|  |  |

Submitted On:

Date: 17/12/2024

**Task No 01: Write a Python program that implements range – kutta methods of the orders mentioned against each part to approximate the 1 st order derivative of the following:**

**a. approximate y(0.2) for 𝒚′ = 𝒙−𝒚 , having x0 = 0, y0 = 1, and h = 0.1. (2nd order) 𝟐**

**b. approximate y(0.3) for 𝒚′ = 𝟏 − 𝒚𝟑, having x0 = 0, y0 = 0, and h = 0.1. (3rd order)**

**c. approximate y(0.1) for 𝒚′ = −𝟐𝒚 + 𝒙𝟑𝒆−𝟐𝒙, having x0 = 0, y0 = 1, & h = 0.1. (4th order)**

**Solution:**

def runge\_kutta\_2nd\_order(f, x0, y0, h, target\_x):

    x, y = x0, y0

    while x < target\_x:

        k1 = h \* f(x, y)

        k2 = h \* f(x + h, y + k1)

        y += 0.5 \* (k1 + k2)

        x += h

    return y

def runge\_kutta\_3rd\_order(f, x0, y0, h, target\_x):

    x, y = x0, y0

    while x < target\_x:

        k1 = h \* f(x, y)

        k2 = h \* f(x + h/2, y + k1/2)

        k3 = h \* f(x + h, y - k1 + 2 \* k2)

        y += (k1 + 4 \* k2 + k3) / 6

        x += h

    return y

def runge\_kutta\_4th\_order(f, x0, y0, h, target\_x):

    x, y = x0, y0

    while x < target\_x:

        k1 = h \* f(x, y)

        k2 = h \* f(x + h/2, y + k1/2)

        k3 = h \* f(x + h/2, y + k2/2)

        k4 = h \* f(x + h, y + k3)

        y += (k1 + 2 \* k2 + 2 \* k3 + k4) / 6

        x += h

    return y

**# Problem (a)**

f\_a = lambda x, y: (x - y) / 2

result\_a = runge\_kutta\_2nd\_order(f\_a, x0=0, y0=1, h=0.1, target\_x=0.2)

print(f"Result for (a): y(0.2) = {result\_a:.4f}")

**# Problem (b)**

f\_b = lambda x, y: 1 - y\*\*3

result\_b = runge\_kutta\_3rd\_order(f\_b, x0=0, y0=0, h=0.1, target\_x=0.3)

print(f"Result for (b): y(0.3) = {result\_b:.4f}")

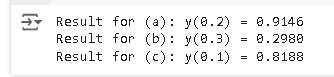
**# Problem (c)**

f\_c = lambda x, y: -2\*y + x\*\*3 \* (2.71828\*\*(-2\*x))  # Approximation of e

result\_c = runge\_kutta\_4th\_order(f\_c, x0=0, y0=1, h=0.1, target\_x=0.1)

print(f"Result for (c): y(0.1) = {result\_c:.4f}")

**Output:**

****