# As1

February 12, 2024

# 1 Assignment 1

### 1.0.1 Chandan Kumar Sahu

### 1.0.2 Roll No. 1911055

[ ]:
```python
import math
import matplotlib.pyplot as plt
from Library import *
```

# 2 Question 1

### 2.0.1 Solve $\exp(-x) - x = 0$ using fixed-point method, accurate up to 4 places in decimal.

[ ]:
```python
# Define the function g(x), this has to be input by the user
def g1(x):
    return math.exp(-x)

initial_guess = 1.0
eps=1e-6

root, Num_iter = fixed_point_method(g1, initial_guess, eps)

print(f"Root of the given equation: {ROUND(root, 4)}")
print(f"Number of iterations performed: {Num_iter}")
```

```
Root of the given equation: 0.5671
Number of iterations performed: 25
```

# 3 Question 2

### 3.0.1 Use Simpson's rule and appropriate Gaussian quadrature to evaluate the following integral accurate up to 6 places in decimal.

$$\int_0^1 \sqrt{1+x^4}dx$$

[ ]:
```python
# SIMPSON'S RULE

# Define the function f(x), this has to be input by the user
def f2(x):
    return math.sqrt(1 + x**4)

a = 0
b = 1
eps = 1e-7

Num_iter_simpson = calculate_N_s(f2, a, b, eps)
integral_simpson = int_simpson(f2, a, b, eps)
print(f"Value of the integral by Simpson's rule: {ROUND(integral_simpson, 6)}")
print(f"Number of iterations performed: {Num_iter_simpson}")
```

```
Value of the integral by Simpson's rule: 1.089429
Number of iterations performed: 30
```

[ ]:
```python
# GAUSSIAN QUADRATURE

integral_gaussian, Num_iter_gaussian = Gaussian_quadrature(f2, a, b, eps)
print(f"Value of the integral by Gaussian quadrature: {ROUND(integral_gaussian,␣
 ↪6)}")
print(f"Number of iterations performed (order of P_n(x)): {Num_iter_gaussian}")
```

```
Value of the integral by Gaussian quadrature: 1.089429
Number of iterations performed (order of P_n(x)): 6
```

# 4 Question 3

#### 4.0.1 Solve the following ODE with RK4 with interval sizes 0.5, 0.2, 0.05 and 0.01. Tabulate your results.

$$y' = \frac{5x^2 - y}{\exp(x+y)} \quad \text{and} \quad y(0) = 1.0$$

[ ]:
```python
def dydx(x, y):
    return (5*x**2-y)/(math.exp(x+y))

x0 = 0.0
xn = 5.0
y0 = 1.0

h1 = 0.5
h2 = 0.2
h3 = 0.05
h4 = 0.01
```

```python
X1, Y1 = ODE_1D_RK4(dydx, y0, x0, xn, h1)
X2, Y2 = ODE_1D_RK4(dydx, y0, x0, xn, h2)
X3, Y3 = ODE_1D_RK4(dydx, y0, x0, xn, h3)
X4, Y4 = ODE_1D_RK4(dydx, y0, x0, xn, h4)
```

[ ]:
```python
# Print the results
print("X1      Y1")
for order in range(len(X1)):
    print(f"{X1[order]:.2f}    {Y1[order]:.4f}")
print()

print("X2      Y2")
for order in range(len(X2)):
    print(f"{X2[order]:.2f}    {Y2[order]:.4f}")
print()

print("X3      Y3")
for order in range(len(X3)):
    print(f"{X3[order]:.2f}    {Y3[order]:.4f}")
print()

print("X4      Y4")
for order in range(len(X4)):
    print(f"{X4[order]:.2f}    {Y4[order]:.4f}")
print()
```

```
X1      Y1
0.00    1.0000
0.50    0.9132
1.00    1.0719
1.50    1.3498
2.00    1.6191
2.50    1.8382
3.00    2.0055
3.50    2.1298
4.00    2.2208
4.50    2.2868
5.00    2.3343

X2      Y2
0.00    1.0000
0.20    0.9378
0.40    0.9104
0.60    0.9267
0.80    0.9838
1.00    1.0716
```

| | |
|---|---|
| 1.20 | 1.1778 |
| 1.40 | 1.2920 |
| 1.60 | 1.4064 |
| 1.80 | 1.5162 |
| 2.00 | 1.6189 |
| 2.20 | 1.7131 |
| 2.40 | 1.7986 |
| 2.60 | 1.8754 |
| 2.80 | 1.9442 |
| 3.00 | 2.0053 |
| 3.20 | 2.0596 |
| 3.40 | 2.1077 |
| 3.60 | 2.1502 |
| 3.80 | 2.1876 |
| 4.00 | 2.2206 |
| 4.20 | 2.2497 |
| 4.40 | 2.2751 |
| 4.60 | 2.2975 |
| 4.80 | 2.3171 |
| 5.00 | 2.3342 |

| X3 | Y3 |
|---|---|
| 0.00 | 1.0000 |
| 0.05 | 0.9821 |
| 0.10 | 0.9656 |
| 0.15 | 0.9507 |
| 0.20 | 0.9378 |
| 0.25 | 0.9271 |
| 0.30 | 0.9189 |
| 0.35 | 0.9133 |
| 0.40 | 0.9104 |
| 0.45 | 0.9104 |
| 0.50 | 0.9131 |
| 0.55 | 0.9185 |
| 0.60 | 0.9267 |
| 0.65 | 0.9375 |
| 0.70 | 0.9507 |
| 0.75 | 0.9662 |
| 0.80 | 0.9838 |
| 0.85 | 1.0034 |
| 0.90 | 1.0246 |
| 0.95 | 1.0474 |
| 1.00 | 1.0716 |
| 1.05 | 1.0969 |
| 1.10 | 1.1231 |
| 1.15 | 1.1502 |
| 1.20 | 1.1778 |
| 1.25 | 1.2060 |

| | |
|---|---|
| 1.30 | 1.2344 |
| 1.35 | 1.2631 |
| 1.40 | 1.2920 |
| 1.45 | 1.3208 |
| 1.50 | 1.3495 |
| 1.55 | 1.3780 |
| 1.60 | 1.4064 |
| 1.65 | 1.4344 |
| 1.70 | 1.4621 |
| 1.75 | 1.4894 |
| 1.80 | 1.5162 |
| 1.85 | 1.5426 |
| 1.90 | 1.5686 |
| 1.95 | 1.5940 |
| 2.00 | 1.6189 |
| 2.05 | 1.6433 |
| 2.10 | 1.6671 |
| 2.15 | 1.6904 |
| 2.20 | 1.7131 |
| 2.25 | 1.7353 |
| 2.30 | 1.7569 |
| 2.35 | 1.7780 |
| 2.40 | 1.7986 |
| 2.45 | 1.8186 |
| 2.50 | 1.8381 |
| 2.55 | 1.8570 |
| 2.60 | 1.8754 |
| 2.65 | 1.8934 |
| 2.70 | 1.9108 |
| 2.75 | 1.9277 |
| 2.80 | 1.9442 |
| 2.85 | 1.9601 |
| 2.90 | 1.9756 |
| 2.95 | 1.9907 |
| 3.00 | 2.0053 |
| 3.05 | 2.0195 |
| 3.10 | 2.0333 |
| 3.15 | 2.0467 |
| 3.20 | 2.0596 |
| 3.25 | 2.0722 |
| 3.30 | 2.0844 |
| 3.35 | 2.0962 |
| 3.40 | 2.1077 |
| 3.45 | 2.1188 |
| 3.50 | 2.1296 |
| 3.55 | 2.1401 |
| 3.60 | 2.1502 |
| 3.65 | 2.1600 |

| | |
|---|---|
| 3.70 | 2.1695 |
| 3.75 | 2.1787 |
| 3.80 | 2.1876 |
| 3.85 | 2.1963 |
| 3.90 | 2.2047 |
| 3.95 | 2.2128 |
| 4.00 | 2.2206 |
| 4.05 | 2.2282 |
| 4.10 | 2.2356 |
| 4.15 | 2.2427 |
| 4.20 | 2.2496 |
| 4.25 | 2.2563 |
| 4.30 | 2.2628 |
| 4.35 | 2.2691 |
| 4.40 | 2.2751 |
| 4.45 | 2.2810 |
| 4.50 | 2.2867 |
| 4.55 | 2.2922 |
| 4.60 | 2.2975 |
| 4.65 | 2.3026 |
| 4.70 | 2.3076 |
| 4.75 | 2.3124 |
| 4.80 | 2.3171 |
| 4.85 | 2.3216 |
| 4.90 | 2.3259 |
| 4.95 | 2.3301 |
| 5.00 | 2.3342 |
| 5.05 | 2.3382 |

| X4 | Y4 |
|---|---|
| 0.00 | 1.0000 |
| 0.01 | 0.9963 |
| 0.02 | 0.9927 |
| 0.03 | 0.9891 |
| 0.04 | 0.9856 |
| 0.05 | 0.9821 |
| 0.06 | 0.9787 |
| 0.07 | 0.9753 |
| 0.08 | 0.9720 |
| 0.09 | 0.9688 |
| 0.10 | 0.9656 |
| 0.11 | 0.9625 |
| 0.12 | 0.9594 |
| 0.13 | 0.9564 |
| 0.14 | 0.9535 |
| 0.15 | 0.9507 |
| 0.16 | 0.9480 |
| 0.17 | 0.9453 |

| | |
|---|---|
| 0.18 | 0.9427 |
| 0.19 | 0.9402 |
| 0.20 | 0.9378 |
| 0.21 | 0.9355 |
| 0.22 | 0.9332 |
| 0.23 | 0.9311 |
| 0.24 | 0.9291 |
| 0.25 | 0.9271 |
| 0.26 | 0.9253 |
| 0.27 | 0.9235 |
| 0.28 | 0.9219 |
| 0.29 | 0.9204 |
| 0.30 | 0.9189 |
| 0.31 | 0.9176 |
| 0.32 | 0.9164 |
| 0.33 | 0.9152 |
| 0.34 | 0.9142 |
| 0.35 | 0.9133 |
| 0.36 | 0.9125 |
| 0.37 | 0.9118 |
| 0.38 | 0.9113 |
| 0.39 | 0.9108 |
| 0.40 | 0.9104 |
| 0.41 | 0.9102 |
| 0.42 | 0.9101 |
| 0.43 | 0.9101 |
| 0.44 | 0.9101 |
| 0.45 | 0.9104 |
| 0.46 | 0.9107 |
| 0.47 | 0.9111 |
| 0.48 | 0.9116 |
| 0.49 | 0.9123 |
| 0.50 | 0.9131 |
| 0.51 | 0.9139 |
| 0.52 | 0.9149 |
| 0.53 | 0.9160 |
| 0.54 | 0.9172 |
| 0.55 | 0.9185 |
| 0.56 | 0.9200 |
| 0.57 | 0.9215 |
| 0.58 | 0.9231 |
| 0.59 | 0.9249 |
| 0.60 | 0.9267 |
| 0.61 | 0.9287 |
| 0.62 | 0.9307 |
| 0.63 | 0.9329 |
| 0.64 | 0.9351 |
| 0.65 | 0.9375 |

| | |
|---|---|
| 0.66 | 0.9399 |
| 0.67 | 0.9425 |
| 0.68 | 0.9451 |
| 0.69 | 0.9478 |
| 0.70 | 0.9507 |
| 0.71 | 0.9536 |
| 0.72 | 0.9566 |
| 0.73 | 0.9597 |
| 0.74 | 0.9629 |
| 0.75 | 0.9662 |
| 0.76 | 0.9695 |
| 0.77 | 0.9730 |
| 0.78 | 0.9765 |
| 0.79 | 0.9801 |
| 0.80 | 0.9838 |
| 0.81 | 0.9876 |
| 0.82 | 0.9914 |
| 0.83 | 0.9953 |
| 0.84 | 0.9993 |
| 0.85 | 1.0034 |
| 0.86 | 1.0075 |
| 0.87 | 1.0117 |
| 0.88 | 1.0159 |
| 0.89 | 1.0202 |
| 0.90 | 1.0246 |
| 0.91 | 1.0291 |
| 0.92 | 1.0336 |
| 0.93 | 1.0381 |
| 0.94 | 1.0428 |
| 0.95 | 1.0474 |
| 0.96 | 1.0522 |
| 0.97 | 1.0569 |
| 0.98 | 1.0618 |
| 0.99 | 1.0667 |
| 1.00 | 1.0716 |
| 1.01 | 1.0765 |
| 1.02 | 1.0816 |
| 1.03 | 1.0866 |
| 1.04 | 1.0917 |
| 1.05 | 1.0969 |
| 1.06 | 1.1020 |
| 1.07 | 1.1073 |
| 1.08 | 1.1125 |
| 1.09 | 1.1178 |
| 1.10 | 1.1231 |
| 1.11 | 1.1285 |
| 1.12 | 1.1338 |
| 1.13 | 1.1393 |

| | |
|---|---|
| 1.14 | 1.1447 |
| 1.15 | 1.1502 |
| 1.16 | 1.1556 |
| 1.17 | 1.1612 |
| 1.18 | 1.1667 |
| 1.19 | 1.1722 |
| 1.20 | 1.1778 |
| 1.21 | 1.1834 |
| 1.22 | 1.1890 |
| 1.23 | 1.1947 |
| 1.24 | 1.2003 |
| 1.25 | 1.2060 |
| 1.26 | 1.2116 |
| 1.27 | 1.2173 |
| 1.28 | 1.2230 |
| 1.29 | 1.2287 |
| 1.30 | 1.2344 |
| 1.31 | 1.2402 |
| 1.32 | 1.2459 |
| 1.33 | 1.2516 |
| 1.34 | 1.2574 |
| 1.35 | 1.2631 |
| 1.36 | 1.2689 |
| 1.37 | 1.2747 |
| 1.38 | 1.2804 |
| 1.39 | 1.2862 |
| 1.40 | 1.2920 |
| 1.41 | 1.2977 |
| 1.42 | 1.3035 |
| 1.43 | 1.3092 |
| 1.44 | 1.3150 |
| 1.45 | 1.3208 |
| 1.46 | 1.3265 |
| 1.47 | 1.3323 |
| 1.48 | 1.3380 |
| 1.49 | 1.3437 |
| 1.50 | 1.3495 |
| 1.51 | 1.3552 |
| 1.52 | 1.3609 |
| 1.53 | 1.3666 |
| 1.54 | 1.3723 |
| 1.55 | 1.3780 |
| 1.56 | 1.3837 |
| 1.57 | 1.3894 |
| 1.58 | 1.3951 |
| 1.59 | 1.4007 |
| 1.60 | 1.4064 |
| 1.61 | 1.4120 |

| | |
|---|---|
| 1.62 | 1.4176 |
| 1.63 | 1.4232 |
| 1.64 | 1.4288 |
| 1.65 | 1.4344 |
| 1.66 | 1.4399 |
| 1.67 | 1.4455 |
| 1.68 | 1.4510 |
| 1.69 | 1.4566 |
| 1.70 | 1.4621 |
| 1.71 | 1.4676 |
| 1.72 | 1.4730 |
| 1.73 | 1.4785 |
| 1.74 | 1.4839 |
| 1.75 | 1.4894 |
| 1.76 | 1.4948 |
| 1.77 | 1.5002 |
| 1.78 | 1.5055 |
| 1.79 | 1.5109 |
| 1.80 | 1.5162 |
| 1.81 | 1.5215 |
| 1.82 | 1.5268 |
| 1.83 | 1.5321 |
| 1.84 | 1.5374 |
| 1.85 | 1.5426 |
| 1.86 | 1.5479 |
| 1.87 | 1.5531 |
| 1.88 | 1.5582 |
| 1.89 | 1.5634 |
| 1.90 | 1.5686 |
| 1.91 | 1.5737 |
| 1.92 | 1.5788 |
| 1.93 | 1.5839 |
| 1.94 | 1.5889 |
| 1.95 | 1.5940 |
| 1.96 | 1.5990 |
| 1.97 | 1.6040 |
| 1.98 | 1.6090 |
| 1.99 | 1.6140 |
| 2.00 | 1.6189 |
| 2.01 | 1.6238 |
| 2.02 | 1.6287 |
| 2.03 | 1.6336 |
| 2.04 | 1.6384 |
| 2.05 | 1.6433 |
| 2.06 | 1.6481 |
| 2.07 | 1.6529 |
| 2.08 | 1.6576 |
| 2.09 | 1.6624 |

| | |
|---|---|
| 2.10 | 1.6671 |
| 2.11 | 1.6718 |
| 2.12 | 1.6765 |
| 2.13 | 1.6811 |
| 2.14 | 1.6858 |
| 2.15 | 1.6904 |
| 2.16 | 1.6950 |
| 2.17 | 1.6995 |
| 2.18 | 1.7041 |
| 2.19 | 1.7086 |
| 2.20 | 1.7131 |
| 2.21 | 1.7176 |
| 2.22 | 1.7221 |
| 2.23 | 1.7265 |
| 2.24 | 1.7309 |
| 2.25 | 1.7353 |
| 2.26 | 1.7397 |
| 2.27 | 1.7440 |
| 2.28 | 1.7484 |
| 2.29 | 1.7527 |
| 2.30 | 1.7569 |
| 2.31 | 1.7612 |
| 2.32 | 1.7654 |
| 2.33 | 1.7697 |
| 2.34 | 1.7739 |
| 2.35 | 1.7780 |
| 2.36 | 1.7822 |
| 2.37 | 1.7863 |
| 2.38 | 1.7904 |
| 2.39 | 1.7945 |
| 2.40 | 1.7986 |
| 2.41 | 1.8026 |
| 2.42 | 1.8066 |
| 2.43 | 1.8106 |
| 2.44 | 1.8146 |
| 2.45 | 1.8186 |
| 2.46 | 1.8225 |
| 2.47 | 1.8264 |
| 2.48 | 1.8303 |
| 2.49 | 1.8342 |
| 2.50 | 1.8381 |
| 2.51 | 1.8419 |
| 2.52 | 1.8457 |
| 2.53 | 1.8495 |
| 2.54 | 1.8533 |
| 2.55 | 1.8570 |
| 2.56 | 1.8607 |
| 2.57 | 1.8644 |

| | |
|---|---|
| 2.58 | 1.8681 |
| 2.59 | 1.8718 |
| 2.60 | 1.8754 |
| 2.61 | 1.8791 |
| 2.62 | 1.8827 |
| 2.63 | 1.8862 |
| 2.64 | 1.8898 |
| 2.65 | 1.8934 |
| 2.66 | 1.8969 |
| 2.67 | 1.9004 |
| 2.68 | 1.9039 |
| 2.69 | 1.9073 |
| 2.70 | 1.9108 |
| 2.71 | 1.9142 |
| 2.72 | 1.9176 |
| 2.73 | 1.9210 |
| 2.74 | 1.9244 |
| 2.75 | 1.9277 |
| 2.76 | 1.9310 |
| 2.77 | 1.9343 |
| 2.78 | 1.9376 |
| 2.79 | 1.9409 |
| 2.80 | 1.9442 |
| 2.81 | 1.9474 |
| 2.82 | 1.9506 |
| 2.83 | 1.9538 |
| 2.84 | 1.9570 |
| 2.85 | 1.9601 |
| 2.86 | 1.9633 |
| 2.87 | 1.9664 |
| 2.88 | 1.9695 |
| 2.89 | 1.9726 |
| 2.90 | 1.9756 |
| 2.91 | 1.9787 |
| 2.92 | 1.9817 |
| 2.93 | 1.9847 |
| 2.94 | 1.9877 |
| 2.95 | 1.9907 |
| 2.96 | 1.9937 |
| 2.97 | 1.9966 |
| 2.98 | 1.9995 |
| 2.99 | 2.0024 |
| 3.00 | 2.0053 |
| 3.01 | 2.0082 |
| 3.02 | 2.0111 |
| 3.03 | 2.0139 |
| 3.04 | 2.0167 |
| 3.05 | 2.0195 |

| | |
|---|---|
| 3.06 | 2.0223 |
| 3.07 | 2.0251 |
| 3.08 | 2.0278 |
| 3.09 | 2.0306 |
| 3.10 | 2.0333 |
| 3.11 | 2.0360 |
| 3.12 | 2.0387 |
| 3.13 | 2.0414 |
| 3.14 | 2.0440 |
| 3.15 | 2.0467 |
| 3.16 | 2.0493 |
| 3.17 | 2.0519 |
| 3.18 | 2.0545 |
| 3.19 | 2.0571 |
| 3.20 | 2.0596 |
| 3.21 | 2.0622 |
| 3.22 | 2.0647 |
| 3.23 | 2.0672 |
| 3.24 | 2.0697 |
| 3.25 | 2.0722 |
| 3.26 | 2.0747 |
| 3.27 | 2.0771 |
| 3.28 | 2.0796 |
| 3.29 | 2.0820 |
| 3.30 | 2.0844 |
| 3.31 | 2.0868 |
| 3.32 | 2.0892 |
| 3.33 | 2.0916 |
| 3.34 | 2.0939 |
| 3.35 | 2.0962 |
| 3.36 | 2.0986 |
| 3.37 | 2.1009 |
| 3.38 | 2.1032 |
| 3.39 | 2.1054 |
| 3.40 | 2.1077 |
| 3.41 | 2.1100 |
| 3.42 | 2.1122 |
| 3.43 | 2.1144 |
| 3.44 | 2.1166 |
| 3.45 | 2.1188 |
| 3.46 | 2.1210 |
| 3.47 | 2.1232 |
| 3.48 | 2.1253 |
| 3.49 | 2.1275 |
| 3.50 | 2.1296 |
| 3.51 | 2.1317 |
| 3.52 | 2.1338 |
| 3.53 | 2.1359 |

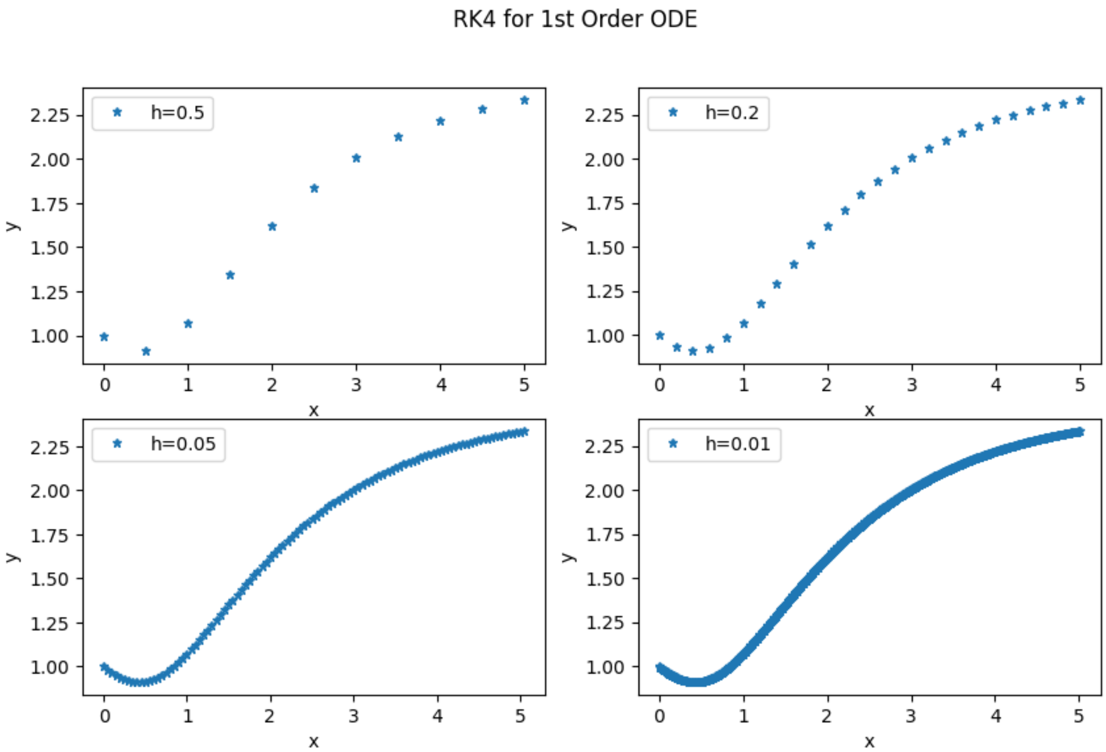
| | |
|---|---|
| 3.54 | 2.1380 |
| 3.55 | 2.1401 |
| 3.56 | 2.1421 |
| 3.57 | 2.1441 |
| 3.58 | 2.1462 |
| 3.59 | 2.1482 |
| 3.60 | 2.1502 |
| 3.61 | 2.1522 |
| 3.62 | 2.1541 |
| 3.63 | 2.1561 |
| 3.64 | 2.1581 |
| 3.65 | 2.1600 |
| 3.66 | 2.1619 |
| 3.67 | 2.1638 |
| 3.68 | 2.1657 |
| 3.69 | 2.1676 |
| 3.70 | 2.1695 |
| 3.71 | 2.1714 |
| 3.72 | 2.1732 |
| 3.73 | 2.1751 |
| 3.74 | 2.1769 |
| 3.75 | 2.1787 |
| 3.76 | 2.1805 |
| 3.77 | 2.1823 |
| 3.78 | 2.1841 |
| 3.79 | 2.1859 |
| 3.80 | 2.1876 |
| 3.81 | 2.1894 |
| 3.82 | 2.1911 |
| 3.83 | 2.1929 |
| 3.84 | 2.1946 |
| 3.85 | 2.1963 |
| 3.86 | 2.1980 |
| 3.87 | 2.1997 |
| 3.88 | 2.2013 |
| 3.89 | 2.2030 |
| 3.90 | 2.2047 |
| 3.91 | 2.2063 |
| 3.92 | 2.2079 |
| 3.93 | 2.2096 |
| 3.94 | 2.2112 |
| 3.95 | 2.2128 |
| 3.96 | 2.2144 |
| 3.97 | 2.2159 |
| 3.98 | 2.2175 |
| 3.99 | 2.2191 |
| 4.00 | 2.2206 |
| 4.01 | 2.2222 |

| | |
|---|---|
| 4.02 | 2.2237 |
| 4.03 | 2.2252 |
| 4.04 | 2.2267 |
| 4.05 | 2.2282 |
| 4.06 | 2.2297 |
| 4.07 | 2.2312 |
| 4.08 | 2.2327 |
| 4.09 | 2.2342 |
| 4.10 | 2.2356 |
| 4.11 | 2.2371 |
| 4.12 | 2.2385 |
| 4.13 | 2.2399 |
| 4.14 | 2.2413 |
| 4.15 | 2.2427 |
| 4.16 | 2.2441 |
| 4.17 | 2.2455 |
| 4.18 | 2.2469 |
| 4.19 | 2.2483 |
| 4.20 | 2.2496 |
| 4.21 | 2.2510 |
| 4.22 | 2.2523 |
| 4.23 | 2.2537 |
| 4.24 | 2.2550 |
| 4.25 | 2.2563 |
| 4.26 | 2.2576 |
| 4.27 | 2.2589 |
| 4.28 | 2.2602 |
| 4.29 | 2.2615 |
| 4.30 | 2.2628 |
| 4.31 | 2.2641 |
| 4.32 | 2.2653 |
| 4.33 | 2.2666 |
| 4.34 | 2.2678 |
| 4.35 | 2.2691 |
| 4.36 | 2.2703 |
| 4.37 | 2.2715 |
| 4.38 | 2.2727 |
| 4.39 | 2.2739 |
| 4.40 | 2.2751 |
| 4.41 | 2.2763 |
| 4.42 | 2.2775 |
| 4.43 | 2.2787 |
| 4.44 | 2.2798 |
| 4.45 | 2.2810 |
| 4.46 | 2.2821 |
| 4.47 | 2.2833 |
| 4.48 | 2.2844 |
| 4.49 | 2.2856 |

| | |
|---|---|
| 4.50 | 2.2867 |
| 4.51 | 2.2878 |
| 4.52 | 2.2889 |
| 4.53 | 2.2900 |
| 4.54 | 2.2911 |
| 4.55 | 2.2922 |
| 4.56 | 2.2932 |
| 4.57 | 2.2943 |
| 4.58 | 2.2954 |
| 4.59 | 2.2964 |
| 4.60 | 2.2975 |
| 4.61 | 2.2985 |
| 4.62 | 2.2996 |
| 4.63 | 2.3006 |
| 4.64 | 2.3016 |
| 4.65 | 2.3026 |
| 4.66 | 2.3036 |
| 4.67 | 2.3046 |
| 4.68 | 2.3056 |
| 4.69 | 2.3066 |
| 4.70 | 2.3076 |
| 4.71 | 2.3086 |
| 4.72 | 2.3095 |
| 4.73 | 2.3105 |
| 4.74 | 2.3115 |
| 4.75 | 2.3124 |
| 4.76 | 2.3134 |
| 4.77 | 2.3143 |
| 4.78 | 2.3152 |
| 4.79 | 2.3162 |
| 4.80 | 2.3171 |
| 4.81 | 2.3180 |
| 4.82 | 2.3189 |
| 4.83 | 2.3198 |
| 4.84 | 2.3207 |
| 4.85 | 2.3216 |
| 4.86 | 2.3225 |
| 4.87 | 2.3233 |
| 4.88 | 2.3242 |
| 4.89 | 2.3251 |
| 4.90 | 2.3259 |
| 4.91 | 2.3268 |
| 4.92 | 2.3276 |
| 4.93 | 2.3285 |
| 4.94 | 2.3293 |
| 4.95 | 2.3301 |
| 4.96 | 2.3310 |
| 4.97 | 2.3318 |

```
4.98    2.3326
4.99    2.3334
5.00    2.3342
5.01    2.3350
```

[ ]:
```python
# Plotting the results
plt.figure(figsize=(10, 6))

plt.subplot(2, 2, 1)
plt.plot(X1, Y1, '*', label='h=0.5', markersize=5)
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
# plt.grid()

plt.subplot(2, 2, 2)
plt.plot(X2, Y2, '*', label='h=0.2', markersize=5)
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
# plt.grid()

plt.subplot(2, 2, 3)
plt.plot(X3, Y3, '*', label='h=0.05', markersize=5)
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
# plt.grid()

plt.subplot(2, 2, 4)
plt.plot(X4, Y4, '*', label='h=0.01', markersize=5)
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
# plt.grid()

plt.suptitle('RK4 for 1st Order ODE')
plt.show()
```




# 5 Question 4

### 5.0.1 Solve the heat equation $u_t = 4u_{xx}$, using Crank-Nicolson and your choice of $\alpha$, subjected to the boundary conditions

$$u(0,t) = 0 = u(8,t) \quad \text{and} \quad x(x,0) = 4x - \frac{x^2}{2}$$

### Since matrix inversion is not taught in class, you may use ready-made available routines for the purpose. Comment on your choice of $\alpha$ and inversion algorithm. Display the solution both in a table and a contour plot.

[ ]:
```
import matplotlib.pyplot as plt

def init_cond(x):
    return 4*x - x**2/2

def plot_diff():
    plt.figure(figsize=(5, 4))
    plt.imshow(solution, extent=[0, L, 0, T], aspect='auto', origin='lower',␣
↪cmap='hot')
    plt.colorbar(label='Temperature')
    plt.title('Heat Diffusion (Crank-Nicolson Method)')
    plt.xlabel('Time')
```

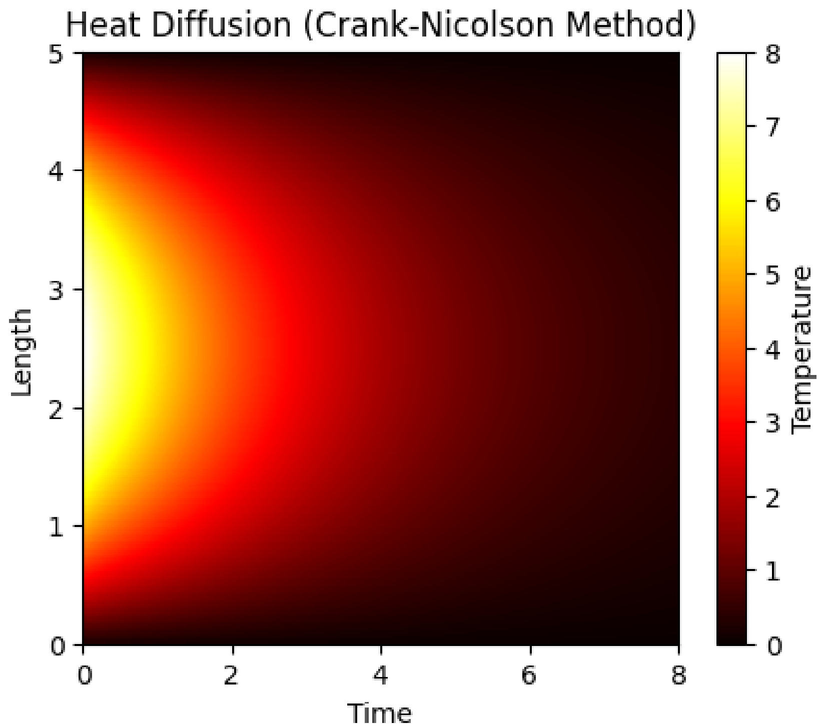
```python
    plt.ylabel('Length')

L = 8.0          # Length of the rod
T = 5.0          # Total time
dx = 0.1         # Spatial step size
dt = 0.01        # Time step size
Diff = 4        # Thermal diffusivity

solution, spatial_grid, time_grid = crank_nicolson_heat_diffusion(L, T, dx, dt,␣
↪Diff, init_cond)
```

Since the Crank-Nicholson method is stable for $\alpha > 0.5$ too, unlike other explicit methods, we can choose any convenient $\alpha$. Here we have taken

$$\alpha = \text{Diff} \times \frac{\Delta t}{\Delta x^2} = 4 \times \frac{0.01}{0.1^2} = 4$$

[ ]:
```python
# Tabulate the solution
print(f"Solution: {solution}")

# Plot the diffusion equation solution
plot_diff()
plt.show()
```

```
Solution: [[0.         0.385      0.24333333 … 0.01679485 0.01669654
0.01659881]
 [0.395      0.5675     0.61833333 … 0.03356504 0.03336857 0.03317326]
 [0.78       0.84625    0.9225     … 0.05028598 0.04999164 0.04969902]
 …
 [0.78       0.84625    0.9225     … 0.05028598 0.04999164 0.04969902]
 [0.395      0.5675     0.61833333 … 0.03356504 0.03336857 0.03317326]
 [0.         0.385      0.24333333 … 0.01679485 0.01669654 0.01659881]]
```




# 6 Question 5

### 6.0.1 Solve the Poisson's equation $u_{xx}+u_{yy} = xe^y$ in a $6^2$ grid with boundary conditions

$$u(0, y) = 0 \quad \text{and} \quad u(2, y) = 2e^y$$
$$u(x, 0) = x \quad \text{and} \quad u(x, 1) = xe$$

### Display the solution both in a table and a 3-D plot.

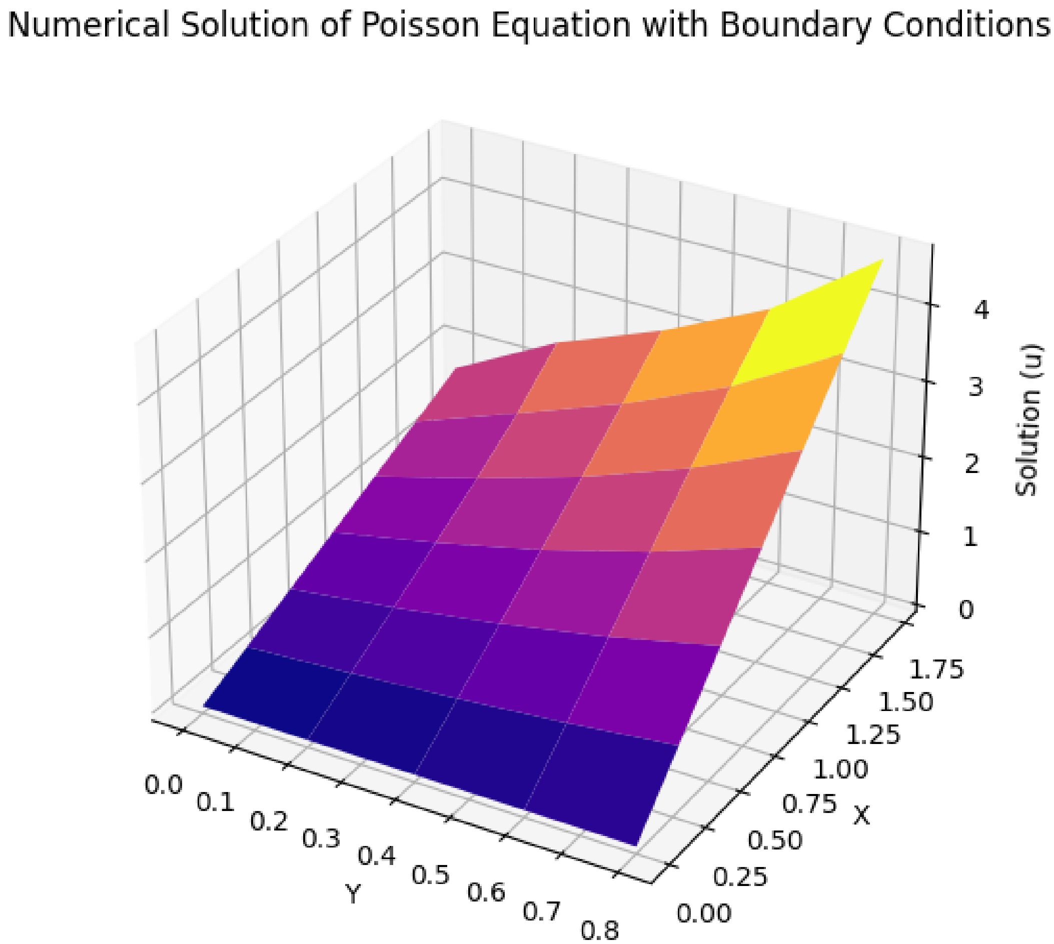
[ ]:
```
def get_BC_poisson(n_x, n_y, x, y):

    # Initial guess
    u = [ [ 0 for j in range(n_y)] for i in range(n_x)]

    # Apply boundary conditions
    for j in range(n_y):
        u[0][j] = 0
        u[-1][j] = 2 * math.exp(y[j])

    for i in range(n_x):
        u[i][0] = x[i]
        u[i][-1] = x[i] * math.exp(1)

    return u
```

```
[ ]: # Set parameters
n_x = 6
n_y = 4
x_length = 2.0
y_length = 1.0

# Solve Poisson equation with boundary conditions
X, Y, u = poisson_eqn_solver(n_x, n_y, x_length, y_length, get_BC_poisson)

# Print the results in matrix like looking form
print("X\tY\tu\t", "X\tY\tu\t", "X\tY\tu\t", "X\tY\tu\t", "X\tY\tu\t",␣
 ↪"X\tY\tu\t")
for j in range(n_y):
    for i in range(n_x):
        print(f"{X[i]:.2f}", end="  ")
        print(f"{Y[j]:.2f}", end="  ")
        print(f"{u[i][j]:.4f}", end="  \t")

    print()
```

```
X       Y       u        X       Y       u        X       Y       u        X
Y       u        X       Y       u        X       Y       u
0.00  0.00  0.0000      0.29  0.00  0.2857      0.57  0.00  0.5714      0.86
0.00  0.8571      1.14  0.00  1.1429      1.43  0.00  1.4286
0.00  0.20  0.0000      0.29  0.20  0.3754      0.57  0.20  0.7518      0.86
0.20  1.1315      1.14  0.20  1.5205      1.43  0.20  1.9371
0.00  0.40  0.0000      0.29  0.40  0.4839      0.57  0.40  0.9688      0.86
0.40  1.4565      1.14  0.40  1.9502      1.43  0.40  2.4561
0.00  0.60  0.0000      0.29  0.60  0.6156      0.57  0.60  1.2317      0.86
0.60  1.8486      1.14  0.60  2.4654      1.43  0.60  3.0750
```

```
[ ]: # Plot the solution
def plot_3D(X, Y, u, colormap):
    fig = plt.figure(figsize=(8, 6))
    ax = fig.add_subplot(111, projection='3d')
    ax.plot_surface(Y, X, u, cmap=colormap)
    ax.set_xlabel('Y')
    ax.set_ylabel('X')
    ax.set_zlabel('Solution (u)')
    ax.set_title('Numerical Solution of Poisson Equation with Boundary␣
 ↪Conditions')

Y, X = np.meshgrid(Y, X)
u = np.array(u)
```

```
[ ]: plot_3D(X, Y, u, 'plasma')
plt.show()
```


Numerical Solution of Poisson Equation with Boundary Conditions
Solution (u)
0
1
2
3
4
X
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
Y
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8

```
[ ]:
```