# PATUAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

**COURSE CODE CCE 312**
**Numerical Methods Sessional**

## SUBMITTED TO:


**Prof. Dr. Md Samsuzzaman**
**Department of Computer and Communication Engineering**
**Faculty of Computer Science and Engineering**


## SUBMITTED BY:


**Md. Sharafat Karim**
ID: **2102024**,
Registration No: **10151**
**Faculty of Computer Science and Engineering**


Assignment 16
Assignment title: Iteration Method
Date of submission: 06 Mon, Oct 2025

## Problem Statement

**Background:**

A pharmacodynamics research team is modeling the concentration of a new heart medication in the bloodstream over time. The drug's effectiveness is directly related to its steady-state concentration, where the rate of administration equals the rate of elimination. The model for the steady-state concentration C (in mg/L) is derived from a nonlinear pharmacokinetic equation.

**The Challenge:**

After simplifying the complex biological model, the team arrives at the following equation that must be solved for the steady-state concentration C:

$C^3 - 4C + 1 = 0$

This equation is transcendental and cannot be solved analytically for an exact solution. The team needs a numerical value for C to determine if the concentration falls within the therapeutic window (the range where the drug is effective but not toxic).

## Code

```python
import numpy as np
import matplotlib.pyplot as plt

def f2(x):
   return x**3 - 4*x + 1

def g2(x):
   return (x**3 + 1) / 4

def iteration_method(g, x, tol=1e-5, max_iter=100):
   for i in range(max_iter):
      x_new = g(x)
      if abs(x_new - x) < tol:
         return x_new
      x = x_new
   raise ValueError(
      "Iteration did not converge within the maximum number of iterations."
   )


root = iteration_method(g2, 1)

# Let's plot the result
x = np.arange(-10, 10, 0.1)

plt.plot(x, f2(x), label='f2(x) = x^3 - 2x + 2')
```

```
plt.scatter(root, f2(root), color='blue')  # Mark the root on the plot
plt.axvline(root, color='purple', linestyle='--', label=f'x = root ({root:.5f})')


plt.axvline(0, color='green', linestyle='--')
plt.axhline(0, color='red', linestyle='--')

plt.xlabel('x')
plt.ylabel('f2(x)')
plt.title(f"Iteration Method Root: {root:.5f}")
plt.grid()
plt.legend()
```

**Visualization**