Attempt 6 Questions

1. Consider the differential equation

\[ \frac{dx}{dt} = ax(t); \quad x(0) = b \]

Define the Euler algorithm to integrate this differential equation and apply this algorithm for \( n \) integration steps each of size \( h \) to evaluate \( x(T = nh) \).

Compare the expression obtained using the Euler algorithm with that obtained when this equation is integrated exactly, and comment on any differences.

2. Write C functions which execute the following tasks:

   (a) Given \( x \), evaluates and returns

   \[ f(x) = \frac{\sqrt{x} \sin(x) + x^5}{(2x - 1)(2x + 5)} \]

   (b) Given \( x \) and \( n \), evaluates the finite Taylor series approximation for \( \log(1 + x) \),

   \[ \sum_{k=1}^{n} (-1)^{k+1} \frac{x^k}{k} \]

   (c) Given two \( 3 \times 3 \) matrices declared double \( a[3][3] \), and double \( b[3][3] \), evaluates and returns the matrix sum of \( a \) and \( b \).
3. Write a C program which reads in \( n \) numbers \( x_1, \ldots, x_n \), and calculates and prints:

(a) The count of values \( x_i \) which are greater than 1.0.

(b) The count of values \( x_i \) which are less than -1.0.

(c) The list of numbers \( x_i \) sorted in increasing order.

4. Consider the following program:

```c
#include <stdio.h>
#include <math.h>
#define N 3
double a[N], y;
main()
{
    int *i_p, *j, k;
double y, *y_p, z[N], *t_p, t;
for ( k = 0; k < N; k++ )
{
    a[k] = 2 * (k+1) / 3;
    z[k] = (k+1)*(k+2);
}
t = a[0] + a[1]/a[2]*a[0] - z[2]*z[1]/z[0];
k = a[0] > a[1] && a[2] == z[0];
y = 5.0;
t_p = &y;
*t_p += 3.0;
z[0] = *t_p;
}
```

Describe the memory map which the declarations in this code generate. For each executable statement, indicate the order of evaluation by adding parenthesis (for example \( x = a + b \rightarrow (x = (a+b)) \)). Trace what happens in memory as the code executes.

5. Describe the Newton method to find the root of a function \( f(x) \). If \( x_i \) is the estimate of the root that this method generates at iteration \( i \), show that the error in this estimate will proportional to \( (x_i - x_0)^2 \) when \( x_i \) is sufficiently close to \( x_0 \), the true root.

Apply this method to find the roots of the function

\[ f(x) = 2x^2 - 5x - 3 \]

correct to three decimal places.
6. Express the following numbers in base 2 (binary), base 10 (decimal), and base 16 (hex): \((124)_{10}, -(f3)_{16}, (7.46)_{10},\) and \((4f.6d)_{16}\).

Assuming integers are stored in 2's complement format using 4 bytes per number, and reals are stored in IEEE single precision format, determine the HEX patterns by which each of these numbers is represented in the computer.

What value does the hex pattern \(dd1d0000\) have if it is interpreted as a 2's complement integer?, and as an IEEE single precision real number?

7. Simpson's Rule to approximate a one dimensional definite integral is given by

\[
\int_{x-h}^{x+h} dy f(y) \approx \frac{h}{3} \left( f(x - h) + 4f(x) + f(x + h) \right)
\]

Prove that the error in this rule is \(O(h^5)\)

Write a C function which implements the extended Simpson's rule to calculate

\[
\int_{a}^{b} dx f(x)
\]

Your C function should have \(a\) and \(b\) as arguments, and should return an estimate for the integral.

8. A sufficient condition for a continuous function \(f(x)\) to have a minimum in an interval \([a, c]\) is that there exists a \(b\) with \(a < b < c\) such that \(f(a) < f(b)\) and \(f(c) < f(b)\). Describe how this observation can be used to define a class of algorithms to find a minimum of \(f(x)\). Show that the optimum algorithm within this class, \(i.e.\) that algorithm which maximises the reduction in uncertainty in the position of the minimum per iteration), makes use of the factor \(R\) which is defined by the equation

\[
\frac{R}{1 - R} = \frac{1 - R}{1}
\]

Find the numerical value of \(R\).

Apply this algorithm to find an estimate of the minimum of the function

\[f(x) = 3x^2 - 2x - 5\]