세계 속의 빛
광주과학기술원

Computer Programming

Lab7

May 6, 2025

Ex1

- **Problem: Newton-Raphson's Method – Looping**
- The Newton-Raphson's Method is a powerful method of finding the roots of the function $f(x)$, that is the values of x for which $f(x) = 0$. Suppose that x_{old} is an estimate of the root. Then a better estimate is x_{new} which is defined by:

$$x_{new} = x_{old} - \frac{f(x_{old})}{g(x_{old})} \quad \text{where } g(x) = \frac{df(x)}{dx}.$$

  This procedure is iterated until convergence is achieved, defined by $|x_{new} - x_{old}| < acc$, where acc is a user-defined accuracy.

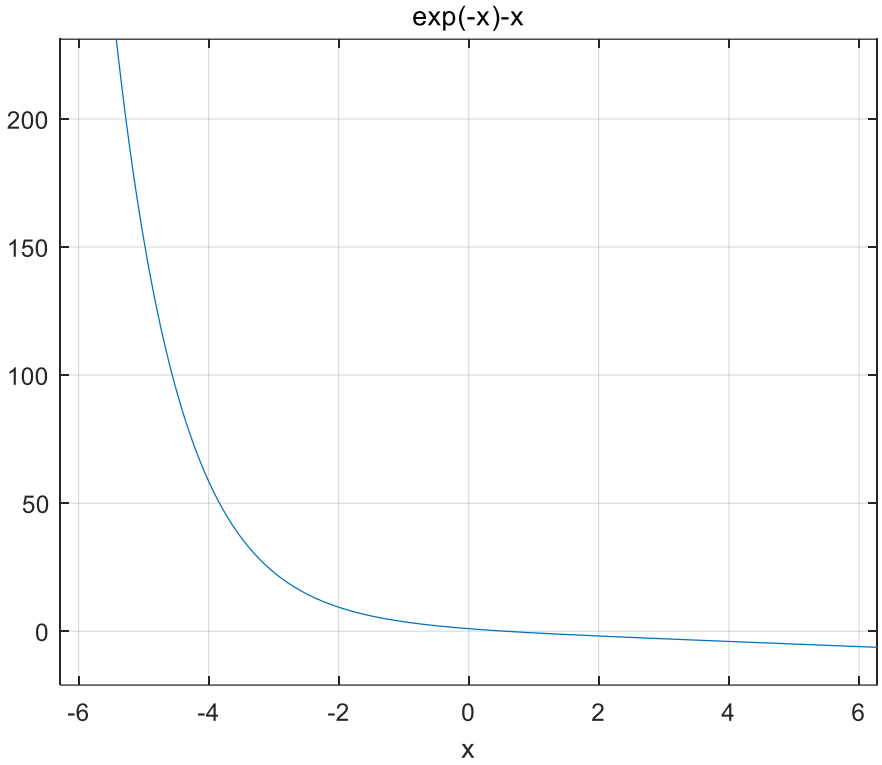
  Use the Newton-Raphson's method to find the root of the function $f(x) = e^{-x} - x$, starting the iteration at $x = -10.0$
- How many iterations are required to obtain an accuracy (acc) of 10^{-6}, and what is the solution at this accuracy?
- Ref
  - https://en.wikipedia.org/wiki/Newton%27s_method

Ex1

- **Program output**

```
[ohyong@cse ~/cp/Lab7]$ vi ex7_1.c
[ohyong@cse ~/cp/Lab7]$ gcc ex7_1.c -o ex7_1 -lm
[ohyong@cse ~/cp/Lab7]$ ./ex7_1
Program to find the root of the function exp(-x)-x using the Newton - Raphson Method.
Enter the accuracy (acc): 0.000001
iteration :  0  =>  x_old :  -10.0000000000      x_new :   -8.9995914192
iteration :  1  =>  x_old :   -8.9995914192      x_new :   -7.9986039096
iteration :  2  =>  x_old :   -7.9986039096      x_new :   -6.9962536491
iteration :  3  =>  x_old :   -6.9962536491      x_new :   -5.9907702695
iteration :  4  =>  x_old :   -5.9907702695      x_new :   -4.9783158361
iteration :  5  =>  x_old :   -4.9783158361      x_new :   -3.9511098789
iteration :  6  =>  x_old :   -3.9511098789      x_new :   -2.8954212482
iteration :  7  =>  x_old :   -2.8954212482      x_new :   -1.7961383785
iteration :  8  =>  x_old :   -1.7961383785      x_new :   -0.6828305410
iteration :  9  =>  x_old :   -0.6828305410      x_new :    0.2107179216
iteration : 10  =>  x_old :    0.2107179216      x_new :    0.5418139234
iteration : 11  =>  x_old :    0.5418139234      x_new :    0.5670263052
iteration : 12  =>  x_old :    0.5670263052      x_new :    0.5671432879
iteration : 13  =>  x_old :    0.5671432879      x_new :    0.5671432904
The root of the function exp(-x)-x at accuracy 0.000001 is 0.567143
The number of iterations is 13
```

Ex1



Ex2

- ***(Taylor Series Approximation of e^x)*** In mathematics, the exponential function e^x can be approximated using an infinite Taylor series expansion:

$$e^x = 1 + \frac{x^1}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots + \frac{x^{n-1}}{(n-1)!}$$

- As the number of terms increases, the approximation becomes more accurate. Write a program that takes a real number x and an integer n, then computes an approximation of e^x using the first n terms of the Taylor series.
- Function prototype
  - double power(int x, int y);
  - double factorial(int n);

Ex2

- **Program output**

```
[ohyong@cse ~/cp/Lab7]$ vi ex7_2.c
[ohyong@cse ~/cp/Lab7]$ gcc ex7_2.c -o ex7_2 -lm
[ohyong@cse ~/cp/Lab7]$ ./ex7_2
Enter the value of x: 2
Enter the number of terms n: 10

Approximated value of e^2.00 => 7.3887125220
Actual value of e^2.00 = 7.3890560989
Absolute error: 0.0003435769
```

```
[ohyong@cse ~/cp/Lab7]$ ./ex7_2
Enter the value of x: 2
Enter the number of terms n: 20

Approximated value of e^2.00 => 7.3890560989
Actual value of e^2.00 = 7.3890560989
Absolute error: 0.0000000000
```

Ex3

- Write a program that generates 10 random numbers between 1 and 100 without using arrays. The program should calculate the sum and average of the generated numbers and print them. The numbers should be generated and processed individually.

Ex3

- **Program output**

```
[ohyong@cse ~/cp/Lab7]$ vi ex7_3.c
[ohyong@cse ~/cp/Lab7]$ gcc ex7_3.c -o ex7_3
[ohyong@cse ~/cp/Lab7]$ ./ex7_3
Random number 1: 72
Random number 2: 80
Random number 3: 67
Random number 4: 20
Random number 5: 55
Random number 6: 54
Random number 7: 47
Random number 8: 32
Random number 9: 57
Random number 10: 17

Sum of random numbers: 501
Average of random numbers: 50.1
```

```
[ohyong@cse ~/cp/Lab7]$ ./ex7_3
Random number 1: 81
Random number 2: 82
Random number 3: 10
Random number 4: 7
Random number 5: 24
Random number 6: 70
Random number 7: 45
Random number 8: 76
Random number 9: 29
Random number 10: 80

Sum of random numbers: 504
Average of random numbers: 50.4
```

Submission

- **Submit to server**

Lab #

Class #

At the end of the Lab7, submit your C sources file by typing

~gs1401/bin/submit Lab7_2 ex7_1.c ex7_2.c ex7_3.c // by Thur. 11:50

~gs1401/bin/submit Lab7_3 ex7_1.c ex7_2.c ex7_3.c // by Friday 10:50

~gs1401/bin/submit Lab7_4 ex7_1.c ex7_2.c ex7_3.c // by Friday 11:50

~gs1401/bin/submit Lab7_5 ex7_1.c ex7_2.c ex7_3.c // by Friday 13:50

You may check that you have submitted your source code correctly by typing

~gs1401/bin/submit -check