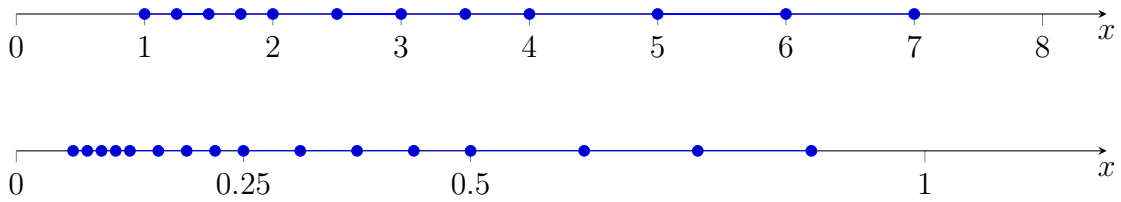
Numerical Method
Purdue University
Spring 2021

Homework 1

Jerry Wang

1 Problem 1

1. The number for all possible numbers in $\mathbb{R}(3, 2)$ is shown below:



2. In the fractional part of the floating point value, the minimum distance is determined by the fractional number represented by the last bit in the mantissa. In the case of this problem, the mantissa has 3 bits, so the smallest difference is $2^3 = 0.125$. The actual distance between the normalized floating point number will multiply the fractional difference with the 2 raised to the exponent. The equation for $d(x)$ is:

$$d(x) = 2^{-3} \cdot 2^e \tag{1}$$

Where: e = exponent of x
The table below shows the distance of all the numbers:

x	$d(x)$	x	$d(x)$
0.0625	0.015625	1	0.25
0.078125	0.015625	1.25	0.25
0.09375	0.015625	1.5	0.25
0.109375	0.015625	1.75	0.25
0.125	0.03125	2	0.5
0.15625	0.03125	2.5	0.5
0.1875	0.03125	3	0.5
0.21875	0.03125	3.5	0.5
0.25	0.0625	4	1
0.3125	0.0625	5	1
0.375	0.0625	6	1
0.4375	0.0625	7	1

3. The table below is the relative distance of all possible floating point value in $\mathbb{R}(3, 2)$

x	$r(x)$	x	$r(x)$
0.0625	1/4	1	1/4
0.078125	1/5	1.25	1/5
0.09375	1/6	1.5	1/6
0.109375	1/7	1.75	1/7
0.125	1/4	2	1/4
0.15625	1/5	2.5	1/5
0.1875	1/6	3	1/6
0.21875	1/7	3.5	1/7
0.25	1/4	4	1/4
0.3125	1/5	5	1/5
0.375	1/6	6	1/6
0.4375	1/7	7	1/7

The upper and lower bound of $r(x)$ is 1/4 and 1/7 respectively.

2 Problem 2

1. In the equation, when computing the square root, the chance of rounding the result is very high. Since $|x|$ is very small, there will be a very significant cancellation errors. The error can be circumvented by computing the result using the following equation instead.
$$f(x) = \frac{(1+x^2)-1}{\sqrt{1+x^2}+1} = \frac{x^2}{\sqrt{1+x^2}+1} \tag{2}$$

2. The expression for the condition number of $f(x)$ is:
$$f(x) = \sqrt{1+x^2} - 1, f'(x) = (1+x^2)^{-1/2}x \tag{3}$$
$$\begin{aligned}(\text{cond } f) &= \left|\frac{xf'(x)}{f(x)}\right| \\ &= \left|\frac{x^2(1+x^2)^{-1/2}}{\sqrt{1+x^2}-1}\right| \\ &= \left|\frac{x^2(\sqrt{1+x^2}+1)}{x^2\sqrt{1+x^2}}\right| \\ &= 1 + \frac{1}{\sqrt{1+x^2}}\end{aligned} \tag{4}$$
If $|x|$ becomes very small, the result of the condition number will be less than 2.

3. Because of the small condition number of $f(x)$, the problem is determined to be well-conditioned. The condition number only consider the error in the input and output of the algorithm. It does not consider the error from algorithmic operations in the algorithm. If the calculation is being done using the equation $f(x) = \sqrt{1+x^2} - 1$, there will be some significant cancellation error. This will make the algorithm ill-conditioned. If the alternative equation is used, which is $f(x) = \frac{x^2}{\sqrt{1+x^2}+1}$, then the problem and algorithm will be well-conditioned.

3 Problem 3

The value of c is calculated using the following equation.

$$c = (\gamma - \gamma_n) \cdot n^d \tag{5}$$

Since the value of c is depended on the value of d, the value of d will be integer value between 1 and 5. The result of the calculation is shown below:

n	γ_n	c (when $d = 1$)	c (when $d = 2$)
2000	0.5774656440682016	-0.49995833333738027	-999.9166666747606
4000	0.5773406596931707	-0.49997916655142305	-1999.9166662056923
6000	0.5772989959200316	-0.4999861109926673	-2999.9166659560037
8000	0.5772781635994182	-0.4999895830826162	-3999.9166646609297
10000	0.5772656640681646	-0.49999166631731207	-4999.9166631731205
12000	0.5772573309894593	-0.499993055117276	-5999.916661407312
14000	0.5772513787620532	-0.4999940472851794	-6999.916661992511
16000	0.5772469145760191	-0.49999479177920136	-7999.916668467222
18000	0.5772434424221	-0.49999537020872786	-8999.916663757102
20000	0.5772406646931927	-0.4999958331963761	-9999.91666392752

From the result table, it shows that the value of c is 0.5 and the value of d is 1. The code for generating the result is show below:

```
n = 1:20000
d = 1:5

sum = 0.0
gamma = 0.57721566490153286
for j = d
    println("\nn          gamma_n                    c")
    global sum = 0.0
    println(j)
    for i = n
        global sum = sum + 1.0/i
        if i % 2000 == 0
            output = sum - log(i)
            c = (gamma - output) * (i ^ j)
            println(i, "    ", output, "    ", c,)
        end
    end
end
```

4 Problem 4

The computed error for both formula is shown in the table below:

k	$\sum_{n=1}^{N}(\frac{1}{n}-\frac{1}{n+1})$ Error	$\sum_{n=1}^{N}(\frac{1}{n(n+1)})$ Error
1	1.1102230246251565e-16	0.0
2	4.440892098500626e-16	3.3306690738754696e-16
3	5.551115123125783e-16	6.661338147750939e-16
4	3.3306690738754696e-16	6.661338147750939e-16
5	1.2656542480726785e-14	1.3100631690576847e-14
6	4.696243394164412e-14	4.7628567756419216e-14
7	1.957323192414151e-13	1.9473311851925246e-13

From the result, the error for both formula increases as the number of term increase. The reason for the increase of error is due to the accumulation of rounding error in each term of the summation. The code for producing the result is shown below:

```
k = 1:7
sum1 = 0.0
sum2 = 0.0

println("k    error1                    error2")
for i = k
    n = 1:10^i
    global sum1 = 0.0
    global sum2 = 0.0
    for j = n
        global sum1 += 1.0/j - 1.0/(j+1)
        global sum2 += 1.0/(j * (j + 1))
    end
    actual = 1 - 1.0/(10^i + 1)
    println(i, "    ", abs(actual-sum1), "    ", abs(actual-sum2))
end
```

5 Problem 5

From the given code snippet, the result of the Hurwitz zeta function is calculated using Riemann zeta function by using the equation below:

$$\begin{aligned} H(s,q) = \sum_{n=0}^{\infty} \frac{1}{(q+n)^s} &= R(s) - \sum_{n=1}^{q} \frac{1}{n^s} \\ &= \left(\sum_{n=1}^{q} \frac{1}{n^s} + \sum_{n=0}^{\infty} \frac{1}{(q+n)^s} \right) - \sum_{n=1}^{q} \frac{1}{n^s} \end{aligned} \tag{6}$$

From equation, the first q terms are canceled from the Riemann zeta function using subtraction. This will create significant cancellation error from the subtraction, since the first q terms in the Riemann zeta function are largest terms in the series. The result of the function in the code snippet will have a larger error as q increases.

6 Problem 6

The inequality of $fl(fl(a+b)+c)) \neq fl(a+fl(b+c))$ can be illustrated using the following condition in Julia

$$a = 1.2, b = 1.0, c = 0.4$$

The result of the Julia program is

$$fl(fl(a+b)+c)) = 2.6, fl(a+fl(b+c)) = 2.5999999999999996$$

which shows that the inequality is true.The order of evaluation result in the smallest error can be proved by first evaluating the result of the machine operation of $a+b+c$. The result can be derived as:

$$\begin{aligned}
\text{fl}(a+b) &= (a+b)(1+\varepsilon_1) \\
\text{fl}(\text{fl}(a+b)+c) &= ((a+b)(1+\varepsilon_1)+c)(1+\varepsilon_{11}) \\
&= (a+b+c+(a+b)\varepsilon_1)(1+\varepsilon_{11}) \\
&= a+b+c+(a+b)\varepsilon_1+(a+b+c)\varepsilon_{11}+\cancelto{0}{(a+b)\varepsilon_1\varepsilon_{11}} \\
&\approx a+b+c+(a+b)\varepsilon_1+(a+b+c)\varepsilon_{11}
\end{aligned}$$

Similarly, the result of other order of operation is shown below:

$$\text{fl}(a+\text{fl}(b+c)) \approx a+b+c+(b+c)\varepsilon_2+(a+b+c)\varepsilon_{22}$$

$$\text{fl}(\text{fl}(a+c)+b) \approx a+b+c+(a+c)\varepsilon_3+(a+b+c)\varepsilon_{33}$$

Since all the value of ε is less than 1eps, the value of error is determined by the coefficient in front of the ε term. For all three cases, it all includes a error term of $(a+b+c)\varepsilon$, so the error for that part is relatively the same between all three order of operations. Since $a > b > c > 0$, we will get $(a+b) > (a+c) > (b+c) > 0$. From the inequality, the rank of the three possible orders of operation from smallest to largest error is:

1. $\text{fl}(a+\text{fl}(b+c)) \leftarrow$ Smallest error
2. $\text{fl}(\text{fl}(a+c)+b)$
3. $\text{fl}(\text{fl}(a+b)+c) \leftarrow$ Largest error

7 Problem 7

From the provided equation, the order of presence for the operation is division, multiplication, and subtraction. The accuracy of the computation on a machine can be derived as:

$$fl(A/B) = (A/B)(1+\varepsilon_1) \tag{7}$$

$$\begin{aligned} fl\left(\frac{A}{B}B\right) &= (A/B)(1+\varepsilon_1)\cdot B\cdot(1+\varepsilon_2) \\ &= A(1+\varepsilon_1+\varepsilon_2+\varepsilon_1\varepsilon_1) \\ &\approx A(1+\varepsilon_1+\varepsilon_2) \end{aligned} \tag{8}$$

$$\begin{aligned} fl\left(A-\frac{A}{B}B\right) &= (A-A(1+\varepsilon_1+\varepsilon_2))(1+\varepsilon_3) \\ &= A(\varepsilon_1+\varepsilon_2)(1+\varepsilon_3) \\ &= A(\varepsilon_1+\varepsilon_1\varepsilon_3+\varepsilon_2+\varepsilon_2\varepsilon_3) \\ &\approx A(\varepsilon_1+\varepsilon_2) \end{aligned} \tag{9}$$

In the derivation, the assumption is that all ε terms are very small, so the value of second order, like $\varepsilon_1\varepsilon_2$, are neglected. The accuracy of the computation is the value of A multiply by the sum of the error from the division and multiplication step which are ε_1 and ε_2 respectively.

8 Problem 8

1. The catastrophic error can happen to the quadratic equation formula when

$$a = 1, b = 200, c = -0.00001$$

The real root of the quadratic equation can be calculated as:

$$\sqrt{b^2 - 4ac} = \sqrt{400^2 - 4 \times 0.00001} = 400.00000005000004...$$

$$x_1 = \frac{-400 + \sqrt{400^2 - 4 \times 0.00001}}{2} = 2.500001983207767 \times 10^{-8}$$

$$x_2 = \frac{-400 - \sqrt{400^2 - 4 \times 0.00001}}{2} = -400.000000025$$

When calculating the root of the quadratic equation using only 10-digits floating point, the result is shown as below:

$$x_1 = \frac{-400 + 400.0000001}{2} = 5.000001124244591 \times 10^{-8}$$

$$x_2 = \frac{-400 - 400.0000001}{2} = -400.00000005000004$$

Compare between the two result, it shows that there are significant error. When $|4ac| \ll b^2$, the result of the square root will be very close the value of b. In the case of calculating x_1, the nominator of the formula is a subtraction between two numbers that are very close to each other, which can leads to catastrophic cancellation error.

2. The two serious flaw of this program is:

   (a) When $p = q = 0$, the result of x_1 is zero, which will cause a divide-by-zero error when calculating x_2.

   (b) If p becomes too large, for example $p = 10^{250}$, there will be overflow when calculating the p^2 term in the square root, since the maximum value for a floating point has an order of 10^{308}.

9 Problem 0

1. I have worked on all the problem on my own except problem 6 which ask our TA, Sai, for some help.

2. The source code is included and the solution is prepare according to the most recent Piazza note.