Chapter 1: Computer Arithmetic

Joseph Sepich Homework

Problem 1

a. Convert the binary numbers to decimal numbers

(i). $(110111001.101011101)_2$

$$1{*}2^8{+}1{*}2^7{+}0{*}2^6{+}1{*}2^5{+}1{*}2^4{+}1{*}2^3{+}0{*}2^2{+}0{*}2^1{+}1{*}2^0{+}1{*}2^{-1}{+}0{*}2^{-2}{+}1{*}2^{-3}{+}0{*}2^{-4}{+}1{*}2^{-5}{+}1{*}2^{-6}{+}1{*}2^{-7}{+}0{*}2^{-8}{+}1{*}2^{-9}$$

$$= 1 * 2^8 + 1 * 2^7 + 1 * 2^5 + 1 * 2^4 + 1 * 2^3 + 1 * 2^0 + 1 * 2^{-1} + 1 * 2^{-3} + 1 * 2^{-5} + 1 * 2^{-6} + 1 * 2^{-7} + 1 * 2^{-9}$$

$$= (441.681640625)_{10}$$

(ii). $(1001100101.01101)_{10}$

$$1{*}2^9{+}0{*}2^8{+}0{*}2^7{+}1{*}2^6{+}1{*}2^5{+}0{*}2^4{+}0{*}2^3{+}1{*}2^2{+}0{*}2^1{+}1{*}2^0{+}0{*}2^{-1}{+}1{*}2^{-2}{+}1{*}2^{-3}{+}0{*}2^{-4}{+}1{*}2^{-5}$$

$$= 1 * 2^9 + 1 * 2^6 + 1 * 2^5 + 1 * 2^2 + 1 * 2^0 + 1 * 2^{-2} + 1 * 2^{-3} + 1 * 2^{-5}$$

$$= (613.40625)_{10}$$

(iii). $(101.01)_2$

$$1 * 2^2 + 0 * 2^1 + 1 * 2^0 + 0 * 2^{-1} + 1 * 2^{-2}$$

$$= 1 * 2^2 + 1 * 2^0 + 1 * 2^{-2}$$

$$= (5.25)_{10}$$

b. Convert the decimal numbers to binary. Keep 10 fractional parts, if needed.

(i). $(100.01)_{10}$

Integer Part

$$100 = 2 * 50 + 0$$

$$50 = 2 * 25 + 0$$

$$25 = 2 * 12 + 1$$

$$12 = 2 * 6 + 0$$

$$6 = 2 * 3 + 0$$

$$3 = 2 * 1 + 1$$

$$1 = 2 * 0 + 1$$

Translates to $(1100100)_2$

Fractional Part

$$0.01 * 2 = 0.02$$
$$0.02 * 2 = 0.04$$
$$0.04 * 2 = 0.08$$
$$0.08 * 2 = 0.16$$
$$0.16 * 2 = 0.32$$
$$0.32 * 2 = 0.64$$
$$0.64 * 2 = 1.28$$
$$0.28 * 2 = 0.56$$
$$0.56 * 2 = 1.12$$
$$0.12 * 2 = 0.24$$

Which gives us the binary number $(1100100.0000001010)_2$

(ii). $(64.625)_{10}$

Integer Part

$$64 = 2 * 32 + 0$$
$$32 = 2 * 16 + 0$$
$$16 = 2 * 8 + 0$$
$$8 = 2 * 4 + 0$$
$$4 = 2 * 2 + 0$$
$$2 = 2 * 1 + 0$$
$$1 = 2 * 0 + 1$$

Translates to $(1000000)_2$

Fractional Part

$$0.625 * 2 = 1.25$$
$$0.25 * 2 = 0.5$$
$$0.5 * 2 = 1.0$$

Which gives us the binary number $(1000000.101)_2$

(iii). $(25)_{10}$

Integer Part

$$25 = 2 * 12 + 1$$
$$12 = 2 * 6 + 0$$
$$6 = 2 * 3 + 0$$
$$3 = 2 * 1 + 1$$
$$1 = 2 * 0 + 1$$

Translates to $(11001)_2$

Fractional part does not exist so we just get the binary number $(11001)_2$

Problem 2

Perform a study on error propogation in the following computation:

$$z = xy$$

Let us define x and y in floating point represetnation:

$$fl(x) = x(1 + \delta_x)$$

$$fl(y) = y(1 + \delta_y)$$

The floating point of z would be defined:

$$fl(z) = z(1 + \delta_z)$$

We can plug in the value of z = xy:

$$\begin{gathered}
fl(z) = fl(x)fl(y)(1 + \delta_z) \\
fl(z) = (x(1 + \delta_x))(y(1 + \delta_y))(1 + \delta_z) \\
fl(z) = (x + x\delta_x)(y + y\delta_y)(1 + \delta_z) \\
fl(z) = (xy + xy\delta_y + yx\delta_x + x\delta_x y\delta_y)(1 + \delta_z) \\
fl(z) = xy + xy\delta_y + yx\delta_x + x\delta_x y\delta_y + \delta_z(xy + xy\delta_y + yx\delta_x + x\delta_x y\delta_y) \\
fl(z) = xy(1 + \delta_y + \delta_x + \delta_y\delta_x + \delta_z(1 + \delta_y + \delta_x + \delta_y\delta_x)) \\
fl(z) = xy + xy(\delta_y + \delta_x + \delta_y\delta_x + \delta_z + \delta_z\delta_y + \delta_z\delta_x + \delta_z\delta_y\delta_x)
\end{gathered}$$

δ_z is the round off error for z. . .

Absolute Error Recall absolute error defined as $fl(x) - x = \delta_x * x$

$$\begin{aligned}
fl(z) - xy &= xy(\delta_y + \delta_x + \delta_y\delta_x + \delta_z + \delta_z\delta_y + \delta_z\delta_x + \delta_z\delta_y\delta_x) \\
&= xy(\delta_y + \delta_x + \delta_y\delta_x) + xy(\delta_z + \delta_z\delta_y + \delta_z\delta_x + \delta_z\delta_y\delta_x)
\end{aligned}$$

Above is the absolute error of fl(z). $xy(\delta_y+\delta_x+\delta_y\delta_x)$ is the propogated error and $xy(\delta_z+\delta_z\delta_y+\delta_z\delta_x+\delta_z\delta_y\delta_x)$ is the round off error.

Relative Error Recall relative error defined as $\delta_x = \frac{fl(x)-x}{x}$

$$\frac{fl(z) - xy}{xy} = \delta_y + \delta_x + \delta_y\delta_x + \delta_z + \delta_z\delta_y + \delta_z\delta_x + \delta_z\delta_y\delta_x$$

Above is the relative error of fl(z). $\delta_y + \delta_x + \delta_y\delta_x$ is the propogated error and $\delta_z + \delta_z\delta_y + \delta_z\delta_x + \delta_z\delta_y\delta_x$ is the round off error.

Problem 3

a. Consider the function:

$$f(x) = \sqrt{x^2+2x+2} - x - 1$$

For what values of x would this function be difficult to compute in a computer? Please explain what difficulty and why. Could you find a way to avoid this difficulty? Explain in detail.

The function could also be written the following way:

$$f(x) = \sqrt{x^2+2x+2} - (x+1) = \sqrt{(x+1)^2+1} - (x+1) = \sqrt{(x+1)^2+1} - \sqrt{(x+1)^2}$$

As you can see in this function you are subtracting very similar values. The only difference between the 2 is the addition of 1 in the first expression before taking the sqaure root. If the number x gets very large (positive), then this could become a problem. As x gets larger and larger. The addition of 1 in the first expression has very neglible effects and you could easily start to lose many significant digits. If x was really big then the computer may even state there is no difference from rounding errors.

$$f(x) = \frac{(\sqrt{x^2+2x+2} - (x+1)) * (\sqrt{x^2+2x+2} + (x+1))}{(\sqrt{x^2+2x+2} + (x+1))}$$

$$f(x) = \frac{x^2+2x+2-(x+1)^2}{(\sqrt{x^2+2x+2} + (x+1))}$$

$$f(x) = \frac{1}{\sqrt{x^2+2x+2} + (x+1)}$$

To solve this problem we can do what we did above and multiply the expression by its conjugate. This changes the function from using subtraction to using addition, so we will no longer lose these significant digits.

b. Explain why the function

$$f(x) = \frac{1}{\sqrt{x+2} - \sqrt{x}}$$

can not be computed accurately in a computer when x is large (using the above formula). Find a way around the problem.

Let's try the conjugate approach that we have been using…

$$f(x) = \frac{\sqrt{x+2} + \sqrt{x}}{(\sqrt{x+2} - \sqrt{x}) * (\sqrt{x+2} + \sqrt{x})}$$

$$f(x) = \frac{\sqrt{x+2} + \sqrt{x}}{x+2+x}$$

$$f(x) = \frac{\sqrt{x+2} + \sqrt{x}}{2x+2}$$

Since we no longer have subtraction of two very close numbers, we will not be the same risk of losing significant digits.

c. Perform the following computations

Use $\frac{1}{3} = 0.333333$, $\frac{3}{4} = 0.75$, $\frac{100}{301} = 0.332226$.

(i). Compute $\frac{1}{3} + \frac{3}{4}$ by using five significant digits rounding arithmetic.

$$0.33333 + 0.75 = 1.0833$$

(ii). Compute $\frac{1}{3} - \frac{100}{301}$ by using 5 significant digits chopping arithmetic.

$$0.33333 - 0.33222 = 0.00111$$

We lose 2 significant digits! (5 to 3, 6 to 4)

Problem 4

a. Derive the following Taylor series for $(1+x)^n$

Write out its particular form at n = 2, n = 3, and n = 1/2. Use the last form to compute $\sqrt{1.001}$ correct to 15 decimal places.

Recall the definition of a Taylor series:

$$f(x) = f(c) + f'(c)(x-c) + \frac{1}{2!}f''(c)(x-c)^2 + ...$$

$$f(x) = \Sigma \frac{f^{(i)}(c)}{i!}(x-c)^i$$

We know the following:

$$f(x) = (1+x)^n$$

And the power rule for derivatives:

$$f'(x) = n(1+x)^{(n-1)}$$

We can then plug in our equation and its derivatives to the form of a taylor series:

Term 0:

$$f(c) = (1+c)^n$$

Term 1:

$$f'(c)(x-c) = n(1+c)^{(n-1)}(x-c)$$

Term 2:

$$f''(c)(x-c)^2/2! = \frac{n(n-1)}{2!}(1+c)^{(n-2)}(x-c)^2$$

If c is 0 we get:

$$f(0) = 1^n = 1$$

$$f'(0)(x-0) = n(1)^{(n-1)}(x) = xn$$

$$f''(0)(x-0)^2/2! = \frac{n(n-1)}{2!}(1)^{(n-2)}(x)^2 = \frac{n(n-1)}{2!}x^2$$

And if you continue this series you get our form:

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

When n = 2:

$$(1+x)^2 = 1 + 2x + x^2 + 0 + ...$$

When n = 3:

$$(1+x)^3 = 1 + 3x + 3x^2 + 2x^3 + ...$$

When n = 1/2:

$$(1+x)^{\frac{1}{2}} = 1 + \frac{1}{2}x + -\frac{1}{8}x^2 + \frac{1}{8}x^3 + ...$$

$\sqrt{1.0001} = (1+0.0001)^{\frac{1}{2}}$ so we can evauluate the series of the form n = 1/2 when x = 0.0001. We are aiming for 15 decimal places.

$$(1+0.0001)^{\frac{1}{2}} = 1 + 0.00005 + -\frac{1}{8}(0.0001)^2 + \frac{1}{8}(0.0001)^3 + ...$$

$$(1+0.0001)^{\frac{1}{2}} = 1 + 0.00005 + -\frac{1}{8}(0.00000001) + \frac{1}{8}(0.000000000001) + ...$$

Since we are multiply anything further(which would be < 1) by $1 * 10^{-16}$, it will not be within 15 decimal places, so we get:

$$(1+0.0001)^{\frac{1}{2}} = 1 + 0.00005 - 0.000000125 + 0.000000000125 = 1.000049875$$

b. Use the answer above to determine a series for $(1+x^2)^{-1}$

If we use a maclaurin series as before, n = 1 and x is now squared. If we plug that into the series then we get...

$$(1+x^2)^n = 1 + x^2n + \frac{n(n-1)}{2!}x^4 + \frac{n(n-1)(n-2)}{3!}x^8 + ...$$

And then plug in n = -1 to get our final answer:

$$(1+x^2)^{-1} = 1 - x^2 + x^4 - x^6 + x^8 - ...$$

Problem 6

a. Write a function to sum the elements of a matrix

```
function sum = MatSum(matrix)
%MATSUM Sums the value of all elements in a matrix
    dims = size(matrix);
    rows = dims(1);
    cols = dims(2);

    sum = 0;
    for row = 1:rows
        for col = 1:cols
           sum = sum + matrix(row,col);
        end
    end

end
```

MatSum([12.2 1.2 2.4; 2 3 4; 2.5 6.2 3.4])

ans =

```
         36.9000
```

b. Write a function to convert decimal to binary

```
function biNum = DecToBin(decNum)
%DECTOBIN Converts the given base 10 number to base 2
%   Calculate integer part, then fractional part, then join together
%   Rounds to a max of 16 digits
    integer = floor(decNum);
    fraction = decNum - integer;

    % integer conversion
    currentNum = integer;
    biNum = {};
    arraySize = 1;
    while currentNum > 0
        remainder = num2str(mod(currentNum, 2));
        biNum = [biNum remainder];
        currentNum = floor(currentNum/2);
    end

    biNum = flip(biNum);

    % fractional conversion
    frac = [];
    decCount = 1;
    while fraction > 0
        if decCount == 16
            break
        end
        fraction = fraction * 2;
        intPart = floor(fraction);
        frac = [frac num2str(intPart)];
        decCount = decCount + 1;
        fraction = fraction - intPart;
    end
    biNum = cellstr(biNum);
    frac = cellstr(frac);

    biNum = strjoin(biNum,'');
    frac = strjoin(frac,'');
    biNum = strcat(biNum,'.');
    biNum = strcat(biNum,frac);
    %biNum = str2double(biNum);
end
```

```
DecToBin(12.625)

ans =

         ‘1100.101’

DecToBin(21.45)

ans =

         ‘10101.011100110011001’
```

Problem 7

a. Write a matlab function called quadroots

```
function [r1, r2] = quadroots(a,b,c)
%QUADROOTS Computes the roots fo a quadratic function
% input: a, b, c; coefficients for the polynomial ax^2+bx+c=0
% output: r1, r2: the two roots for the polynomial
    r1 = ((-1 * b) + sqrt(b^2 - (4 * a * c)))/ (2 * a);
    r2 = ((-1 * b) - sqrt(b^2 - (4 * a * c)))/ (2 * a);
end
```

b.

```
[r1, r2] = quadroots(2,6,-3)

r1 =

         0.4365

r2 =

         -3.4365

[r1, r2] = quadroots(1,-14,49)

r1 =

         7
```

```
r2 =

          7

[r1, r2] = quadroots(3,-123454321,2)

r1 =

          4.1151e+07

r2 =

          1.7385e-08
```

We lose significant digits in the last polynomial. The b coefficient has a very large number and the top of the expression is -b + sqrt(b^2 - 4ac). Since b is so much larger than the other digits, this creates a situation where you are subtracting two very close digits and losing accuracy in significant digits in the process.

c.

```
function [r1, r2] = smartquadroots(a,b,c)
%SMARTQUADROOTS calculates the roots of a polynomial, avoiding loss of
%significance
%  input: a,b,c: coefficients of quadratic polynomial
% output: r1,r2: roots of the given quadratic polynomial
    [r1, r2] = quadroots(a,b,c);

    % Only r2 can see loss of significance because it is the one dealing
    % with subtraction
    if r1 * r2 ~= c/a
        r2 = c/(a * r1);
    end
end
```

```
[r1, r2] = quadroots(3,-123454321,2)

r1 =

          4.1151e+07

r2 =

          1.6200e-08
```