Práctica Dirigida 1
Análisis y Modelamiento Numérico I B

Jimenez Chacon Joel
Vásquez Levano Diego
López Flores Royer Amed
Parishuaña Ortega Jorge Luis
Lique Lamas Alexander Leonardo

Problemas: 1,5,9,13,17

2021-1

April 21, 2021

PD1 Problema 1

Escribir el número decimal correspondiente a los siguientes números

- a) 1101110_2 utilizando el algoritmo hecho en clase

$$b_6 = a_6 = 1$$

$$b_5 = a_5 + 2b_6 = 1 + 2(1) = 3$$
$$b_4 = a_4 + 2b_5 = 0 + 2(3) = 6$$
$$b_3 = a_3 + 2b_4 = 1 + 2(6) = 13$$
$$b_2 = a_2 + 2b_3 = 1 + 2(13) = 27$$
$$b_1 = a_1 + 2b_2 = 1 + 2(27) = 55$$
$$b_0 = a_0 + 2b_1 = 0 + 2(55) = 110$$

finalmente decimos que el número 1101110_2 es 110 en base decimal.

PD1 Problema 1

- c) $100111,101_2$ para la parte entera, análoga al primer item

$$b_5 = a_5 = 1$$

$$b_4 = a_4 + 2b_5 = 0 + 2(1) = 2$$
$$b_3 = a_3 + 2b_4 = 0 + 2(2) = 4$$
$$b_2 = a_2 + 2b_3 = 1 + 2(4) = 9$$
$$b_1 = a_1 + 2b_2 = 1 + 2(9) = 19$$
$$b_0 = a_0 + 2b_1 = 1 + 2(19) = 39$$

para la parte decimal: según el libro guía de la clase, se multiplica a la parte decimal por 1010_2 o 10_{10}, en nuestro caso tendríamos 0.101

$$r_1 = (0.101_2)$$

$$w_1 = (1010_2) * (r_1) = 6.25 \rightarrow b'_1 = 6, r_2 = 0,010$$
$$w_2 = (1010_2) * (r_2) = 2.5 \rightarrow b'_2 = 2, r_3 = 0,100$$
$$w_3 = (1010_2) * (r_3) = 5 \rightarrow b'_3 = 5, r_4 = 0,000$$

por lo tanto $(0.101_2) = 0.625_{10}$ finalmente juntando ambas partes tendríamos que $100111,101_2$ es $39,625_{10}$

PD1 Problema 5

Realice las operaciones aritmética con enteros complemento a dos con una longitud de palabra de $N = 4$ bits para las siguientes operaciones:

- a) $1001 + 0101$
- c) $0011 + 0100$

PD1 Problema 5

Realice las operaciones aritmética con enteros complemento a dos con una longitud de palabra de N=4 bits para las siguientes operaciones:

$$a)1001 + 0101$$

Podemos ver que el numero 1001 tiene un 1 en el bit correspondiente al signo. Para hallar el valor absoluto de este numero negativo, hallamos su complemento a 2 y le agregamos el signo:
Para ello primero invertimos los bits del numero, y luego sumamos 1. Finalmente cambiamos el primer bit por el signo negativo

$$\begin{array}{rlllll} & 1001 & \rightarrow 0110 & \rightarrow \text{-}111 & \rightarrow \text{-}7 \\ + & 0101 & & & \rightarrow 5 \\ \hline & 1110 & \rightarrow 0001 & \rightarrow \text{-}010 & \rightarrow \text{-}2 \end{array}$$

PD1 Problema 5

Realice las operaciones aritmética con enteros complemento a dos con una longitud de palabra de N=4 bits para las siguientes operaciones:

$$c)0011 + 0100$$

En este caso ambos numeros son positivos, por lo que no es necesario calcular su complemento a 2.

$$\begin{array}{rll} & 0011 & \rightarrow 3 \\ + & 0100 & \rightarrow 4 \\ \hline & 0111 & \rightarrow 7 \end{array}$$

PD1 Problema 9

Si tenemos β = 2, t = 3, L = -1 y U = 2.

- a) Determine el intervalo donde se representa los números reales.
- c) Determine los números de máquina que contiene dicho intervalo.

a) Determine el intervalo donde se representa los números reales.

Para el sistema de Punto Flotante $\mathbb{F}(2,3,-1,2)$ se expresarán números de la forma:

$$(-1)^s(0.a_1a_2a_3)_2 * 2^e donde \;\; e \in [-1,2] \;\; y \;\; a_1 \neq 0$$

De esta forma el menor numero, sin contar el signo, que se puede representar es: β^{L-1}

$$x_{min} = 0.100_2 * 2^{-1} = 2^{-2} = \frac{1}{4} = 0.25$$

Mientras que el Mayor será: $\beta^U - \beta^{U-t}$

$$\begin{aligned} x_{max} = 0.111_2 * 2^2 &= 1.000_2 * 2^2 - 0.001_2 * 2^2 \\ &= 2^2 - 2^{-3} * 2^2 \\ &= 2^2 - 2^{-1} \\ &= 4 - 0.5 = 3.5 \end{aligned}$$

a) Determine el intervalo donde se representa los números reales.

Finalmente sabiendo que si $x \in \mathbb{F}(2, 3, -1, 2)$, se cumple:

$$x_{min} \leq |x| \leq x_{max}$$

$$0.25 \leq |x| \leq 3.5$$

Entonces el intervalo que contiene a los numeros reales que pueden ser representados con F es:

$$x \in [-3.5, -0.25] \; U \; [0.25, 3.5] \subset \mathbb{R}$$

c) Determine los números de máquina que contiene dicho intervalo.

c) Para determinar los números maquina del sistema:
Sean los numeros $x \in \mathbb{F}(2,3,-1,2)$ positivos de la forma $(-1)^s m_2 * 2^e$:

$m_2 =$	m_{10}	$2^e =$	2^{-1}	2^0	2^1	2^2
$s = 1$						
$0.100_2 =$	0.5	|	0.25	0.5	1.0	2.0
$0.101_2 =$	0.625	|	0.3125	0.625	1.25	2.5
$0.110_2 =$	0.75	|	0.375	0.75	1.5	3.0
$0.111_2 =$	0.875	|	0.4375	0.875	1.75	3.5
$s = -1$						
$-0.100_2 =$	-0.5	|	-0.25	-0.5	-1.0	-2.0
$-0.101_2 =$	-0.625	|	-0.3125	-0.625	-1.25	-2.5
$-0.110_2 =$	-0.75	|	-0.375	-0.75	-1.5	-3.0
$-0.111_2 =$	-0.875	|	-0.4375	-0.875	-1.75	-3.5

PD1 Problema 13

Realizar la siguiente en Python:

- a) $(1,2-1)-0,2$
- b) $1,2-(1+0,2)$

Podemos concluir que:

- a) La propiedad asociativa en el conjunto $\mathbb{F}$, ¿ Se cumple ?
- b) Explique claramente el motivo de esto.

a) La propiedad asociativa en el conjunto $\mathbb{F}$, ¿ Se cumple ?

```
[2] x = (1.2 - 1) - 0.2
    print(f"El resultado es: {x}")
```

```
El resultado es: -5.551115123125783e-17
```

```
x = 1.2 - (1 + 0.2)
print(f"El resultado es: {x}")
```

```
El resultado es: 0.0
```

¿Es algún error de Python?

El cálculo también fue ejecutado en Javascript, obteniendo el siguiente resultado:

Resultado de las siguientes operaciones:

(1.2 - 1) - 0.2
Resultado: -5.551115123125783e-17

1.2 - (1+0.2)
Resultado: 0

b) Explique claramente el motivo de esto.

Cuando trabajamos con un computador a nosotros nos interesa representar números en formato binario, centrandonos en los números de punto flotante, casos como el 0.5 nos dan de forma exacta una representación en binario pero en el caso de 0.1, no .

Representación matemática de 0.1 en binario

.

- 0.0001100110011001100110011001100110011001100110011...

Python solo imprime una aproximación decimal al valor decimal verdadero de la aproximación binaria almacenada por la máquina.

- 0.1000000000000000055511151231257827021181583404541015625

b) Explique claramente el motivo de esto.

Estaremos usando la función decimal(), para imprimir el valor al cual se esta aproximando el numero ingresado.

```
[7]  from decimal import Decimal

[8]  Decimal(1.2)

     Decimal('1.1999999999999999555910790149937383830547332763671875')

[9]  Decimal(1)

     Decimal('1')

     Decimal(0.2)

     Decimal('0.200000000000000011102230246251565404236316680908203125')
```

b) Explique claramente el motivo de esto.

Los valores aproximados son los siguientes:

Números:

```
- 1.2:

    - aprox: 1.19999999999999995559107901499e0

    - binario: 0 11111111110

                 0011001100110011001100110011001100110011001100110011

- 1:

    - aprox: 1.0e0

    - binario: 0 11111111110

                 0000000000000000000000000000000000000000000000000000

- 0.2:

    - aprox: 2.00000000000000011102230246252e-1

    - binario: 0 111111110

                 001001100110011001100110011001100110011001100110011010
```

b) Explique claramente el motivo de esto.

Realizando los cálculos, usando explicitamente los valores aproximados.

```
[6] x = (1.19999999999999995559107901499e0 - 1.0e0)
    x
```

```
0.19999999999999996
```

```
x - 2.00000000000000011102230246252e-1
```

```
-5.551115123125783e-17
```

b) Explique claramente el motivo de esto.

```
[3] x = (2.00000000000000011102230246252e-1 + 1.0e0)
    x

    1.2

    x - 1.19999999999999995559107901499e0

    0.0
```

PD1 Problema 17

En clase se ha visto distintas formas de sumar:

- a) $sum([0, 1] * 10)$
- b) $math.fsum([0, 1] * 10)$

Explique:

- a) La diferencia de ambos métodos y la diferencia principal.
- b) Explique porque en el primero no se obtiene el valor de 1.

a) La diferencia de ambos métodos y la diferencia principal.

17. En clase se ha visto distintas formas de sumar:

```
[9]  print("sum([0.1]*10) =",sum([0.1]*10))
     print("math.fsum([0.1] * 10) =", math.fsum([0.1] * 10))
```

```
sum([0.1]*10) = 0.9999999999999999
math.fsum([0.1] * 10) = 1.0
```

La diferencia principal es que el método 'sum()' recorre el objeto iterable haciendo una suma uno a uno, sin cuidado alguno respecto a los errores de redondeo, en cambio el metodo 'fsum()' de la libreria 'math' hace una operacion mas cuidadosa, es decir, realiza una suma teniendo en cuenta "los digitos perdidos" en el proceso, asegurando una mayor precision en el proceso de suma.

b) Explique porque en el primero no se obtiene el valor de 1.

Sumemos termino a termino:
Tenemos:

$$0.1 = 1.1001100110011001100110011001100110011001100110011010_2 * 2^{-4}$$

$$0.1 = 1.1001100110011001100110011001100110011001100110011010_2 * 2^{-4}$$

Sumamos:

$$0.1 + 0.1 =$$

$$= 11.0011001100110011001100110011001100110011001100110100_2 * 2^{-4}$$

$$= \ 1.10011001100110011001100110011001100110011001100110100_2 * 2^{-3}$$

$$Redondeando:$$

$$= 1.1001100110011001100110011001100110011001100110011010_2 * 2^{-3}$$

$$Comparando \ \ con \ \ 0.2$$

$$0.2 =$$

$$= 1.1001100110011001100110011001100110011001100110011010_2 * 2^{-3}$$

b) Explique porque en el primero no se obtiene el valor de 1.

Sumemos termino a termino:
Tenemos:

$$0.1 + 0.1 =$$

$$= 1.1001100110011001100110011001100110011001100110011010_2 * 2^{-3}$$

$$0.1 = 0.11001100110011001100110011001100110011001100110011010_2 * 2^{-3}$$

$$\textit{Sumando}:$$

$$0.1 + 0.1 + 0.1 =$$

$$= 10.0110011001100110011001100110011001100110011001100110_2 * 2^{-3}$$

$$= 1.001100110011001100110011001100110011001100110011001110_2 * 2^{-2}$$

$$\textit{Redondeando}:$$

$$= 1.0011001100110011001100110011001100110011001100110100_2 * 2^{-2}$$

$$\textit{Comparando}\ \ \textit{con}\ \ 0.3$$

$$0.3 =$$

$$= 1.0011001100110011001100110011001100110011001100110011_2 * 2^{-2}$$

b) Explique porque en el primero no se obtiene el valor de 1.

Sumemos termino a termino:
Tenemos:

$$0.1 + 0.1 + 0.1 =$$

$$= 1.0011001100110011001100110011001100110011001100110100_2 * 2^{-2}$$

$$0.1 = 0.010011001100110011001100110011001100110011001100110011010_2 * 2^{-2}$$

$$\textit{Sumando}:$$

$$0.1 + 0.1 + 0.1 + 0.1 =$$

$$= 1.100110011001100110011001100110011001100110011001101010_2 * 2^{-2}$$

$$\textit{Redondeando}:$$

$$= 1.1001100110011001100110011001100110011001100110011011_2 * 2^{-2}$$

$$\textit{Comparando con } 0.4$$

$$0.4 =$$

$$= 1.1001100110011001100110011001100110011001100110011010_2 * 2^{-2}$$

b) Explique porque en el primero no se obtiene el valor de 1.

Sumemos termino a termino:
Tenemos:

$$0.1 + 0.1 + 0.1 + 0.1 =$$

$$= 1.1001100110011001100110011001100110011001100110011011_2 * 2^{-2}$$

$$0.1 = 0.011001100110011001100110011001100110011001100110011010_2 * 2^{-2}$$

$$Sumando:$$

$$0.1 + 0.1 + 0.1 + 0.1 + 0.1 =$$

$$= 10.00000000000000000000000000000000000000000000000000110_2.2^{-2}$$

$$Redondeando:$$

$$= 1.0000000000000000000000000000000000000000000000000001_2 * 2^{-1}$$

$$Comparando\ \ con\ \ 0.5$$

$$0.5 =$$

$$= 1.0000000000000000000000000000000000000000000000000000_2 * 2^{-2}$$

b) Explique porque en el primero no se obtiene el valor de 1.

Finalmente tenemos los valores:

$0.1 + 0.1 + 0.1 + 0.1 + 0.1 =$

$= 1.0000000000000000000000000000000000000000000000000001_2 * 2^{-1}$

$0.5 =$

$= 1.0000000000000000000000000000000000000000000000000000_2 * 2^{-2}$

Podemos ver que incluso siendo $0.5 = 0.1_2$ Un numero en base 2 con un numero finito de cifras luego de la coma. Al momento de sumar uno a uno los 0.1 Se tiene un pequeño error.
por lo que al final se va a arrastrar este error que notaremos en el método sum(), error que corrige el metodo fsum(), comprobándose la operación de forma manual como acabamos de hacer