

## Trabajo Práctico N° 1

### SISTEMAS DE NUMERACION

#### OBJETIVOS:

*Elementos que definen un sistema numérico. Conversiones entre sistemas. Diferentes formas de representar números enteros negativos. Sumas y restas de números con y sin signo. Interpretar las funciones de los indicadores de unidades aritméticas (flags). Extensión de signo a  $n$  bits. Representación de números con parte fraccionaria en punto fijo y punto flotante.*

#### A.- SISTEMAS NUMERICOS

1. Definir:
  - 1.1. Sistema numérico posicional y no posicional.
  - 1.2. Base de un sistema numérico.
  - 1.3. Los siguientes sistemas numéricos: decimal, octal, binario, hexadecimal
2. Dado el número 10 en base 16, y 10 en binario, indicar en decimal qué números son. Generalizar para 10 en cualquier base
3. Hallar el siguiente de cada número (expresados en hexadecimal):  
FFFF 2ABF 3B99 1FF C0D0 AOF 999

#### B.- CAMBIOS DE BASE.

4. Discutir los procedimientos utilizados para pasar un número de una base a otra.
5. Convertir al sistema decimal cada uno de los siguientes números en sistema binario:  

1010	1001	1101
1011	1010	
6. Convertir los siguientes números decimales :
  - 6.1. A binario: 45, 318, 319, 5621, 892345, 892346
  - 6.2. A base 16: 592, 2401, 2402.
7. Convertir a binario los siguientes números decimales con parte fraccionaria :  
 435,543 167,761 1024,4201
  - (a) Obtener el resultado en 16 bits e indicar cuál la precisión obtenida
  - (b) Determinar el número necesario de bits para que la precisión sea del 0.1% o superior
8. Cuantos dígitos se necesitan para representar un número de 10 dígitos decimales en las bases binaria y hexadecimal.
9. Encontrar una expresión que determine el número **m** de dígitos requeridos para representar un número N en base **b2** si en la base **b1** se necesitan **n** dígitos.

#### C.- BASES QUE SON POTENCIAS DE OTRAS BASES

10. Sin pasar por el sistema decimal, realizar las siguientes conversiones:
  - a) A la base 2 y 8 el número hexadecimal ABCD,EF
  - b) A base 16 el número binario 111100001
11. Determinar el valor decimal de c/u de los números convertidos en el ejercicio anterior

**D.- OPERACIONES EN LAS DISTINTAS BASES**

12. Construir las tablas de sumar y multiplicar en las bases 2 y 16.

13. Indicar si las siguientes sumas son correctas en alguna base:

$$6 + 7 = 11$$

$$5 + 7 = 13$$

$$5 + 7 = 17$$

**E.- COMPLEMENTO DE UN NUMERO: Complemento a la base.**

14. Justificar el concepto y la necesidad de módulo y complemento de un número.

15. Escribir en decimal el complemento 2 de los siguientes números decimales utilizando 4 dígitos: 0, 1, 10, 32, 65, 90, 98, 99, 100, 128, 4998

16. Un procesador opera con números de 8 bits. En un programa se realizan operaciones con signo. Escribir en decimal los siguientes números:

RA = 11111010 RB = 11111111 RC = 00000000 RD = 10000000 RE = 00000001 RF = 01110101  
RH = 10000001 RL = 01111111;

Considerar que los números están expresados en las convenciones de:

- Magnitud y signo.
- Complemento a la base
- Complemento a la base menos uno

17. Cuales son el mayor número y el menor que se puede escribir con 8 bits en las convenciones de complemento a la base, a la base menos uno, con valor absoluto y sin signo.

**F.- OPERACIONES ARITMETICAS Y LOS INDICADORES DE LAS MISMAS.**

18. Explicar el concepto y la necesidad de disponer de indicadores (flags) en un procesador que opera con registros de n bits. Justificar la existencia de los indicadores de arrastre (carry), desborde (overflow), cero, signo, paridad.

19. Realizar, previa conversión al sistema binario las siguientes operaciones, en las que los números están expresados sin signo:

$$190+260$$

$$450+579$$

20. Realizar, previa conversión al sistema binario las siguientes operaciones, considerando que los números expresados se representan con signo (Trabajar en 6 bits en CM y en CM-1):

$$26+19$$

$$26+32$$

$$26-19$$

$$26-26$$

$$19-26$$

$$-26+19$$

$$-26+26$$

$$-19+26$$

$$-19-26$$

$$-19-30$$

$$-19-31$$

$$-19-32.$$

- Verificar, mediante el análisis de los indicadores, si las operaciones producen un resultado correcto.
- Expresar los resultados también en base 10.

21. Interpretar los registros del problema 16 como (a) enteros sin signo (b) complemento a la 2. Efectuar las siguientes sumas y para cada suma, tanto para el caso (a) como el caso(b), indicar:

- el estado de todos los flags luego de efectuada la operación.
- el resultado en base 10

- analice los flags para justificar si el resultado es correcto o se fue de rango

RA+RA	RA+RH	RA+RD	RL+RE	RL+RD	RB+RH
RF+RH	RC+RL	RE+RF	RB+RC		

22. Se tienen los siguientes dos números binarios expresados en complemento a 2 y definidos con diferente cantidad de bits.

100010      10101011

Efectuar su suma.

23. La medición de tiempo no utiliza la misma base en cada dígito. Calcule el peso de cada posición e indique una metodología para realizar sumas y restas con esta representación.
24. Una unidad aritmético-lógica realizó la siguiente suma  $10110001 + 11100111$  para restar dos números. Indicar qué números decimales restados originaron dicha suma a) Suponiendo que fueron enteros b) Suponiendo que fueron naturales.

### G.- PUNTO FLOTANTE

25. Discutir la necesidad de la representación en punto flotante y posibles formas de implementación. Norma IEEE.
26. Obtener el rango de valores reales representados en punto flotante para las normas IEEE simple precisión.
27. Convertir a punto flotante (convención IEEE de simple precisión) los números decimales  
-2149.35      1926      83,1      0,0022
28. Dado el siguiente número en punto flotante (convención IEEE). expresado en hexadecimal C28FFF00, indicar qué número es en base 10.
29. Determinar cuántos dígitos decimales de precisión se obtienen en la representación en punto flotante del IEEE de simple y doble precisión
30. Discutir cuales son los pasos a seguir para obtener la suma de dos números en punto flotante con resultado normalizado. Obtener la suma de :  $(- 0.13567 \times 10^{+3}) + (+0.67430 \times 10^{-1})$

### H.- EFECTOS DEL FORMATO NUMÉRICO SOBRE EL RESULTADO DE CÓDIGO DE ALTO NIVEL

*Programas escritos en lenguajes de alto nivel pueden tener comportamientos no esperados si no se tiene en cuenta el formato con el cual las variables son almacenadas bit a bit.*

31. Respecto del código detallado más abajo contestar las siguientes preguntas:
- Antes de correr la aplicación, ¿Qué resultado esperaba obtener por pantalla?
  - ¿Qué resultado obtuvo?
  - Explique detalladamente por qué se obtuvo ese resultado.
  - ¿Qué haría para obtener el resultado que esperaba?

```

#include <stdio.h>
int main(int argc, char** argv){
    float puntoFlotante;
    //Prueba 1:
    printf("PRUEBA 1: \n\n");
    puntoFlotante = 276.2546;
    printf("%f \n", puntoFlotante);
    //-----
    //Prueba 2:
    printf("\nPRUEBA 2: \n\n");
    puntoFlotante = 0.8;
    printf("%f \n", puntoFlotante);
    if(puntoFlotante < 0.8)
        printf("Es menor \n");
    else if(puntoFlotante == 0.8)
        printf("Es igual \n");
    else
        printf("Es mayor \n");
    //-----
    //Prueba 3:
    printf("\nPRUEBA 3: \n\n");
    puntoFlotante = 4000.25;
    printf("%f \n", puntoFlotante);
    puntoFlotante += 0.005;
    printf("%f \n", puntoFlotante);
    //-----
}

```

**32.** Dado el siguiente código en C

```

#include <stdio.h>
int main()
{
    int a = -3;
    unsigned int b = 3;
    if(a < b)
        printf("a es menor a b");
    else if(a > b)
        printf("a es mayor a b");
    else
        printf("a es igual a b");
}

```

y sabiendo que los enteros ocupan 4bytes, se pide explicar detalladamente por qué se obtuvo por pantalla “*a es mayor a b*” en lugar del resultado esperado “*a es menor a b*”.

**33.** Dado el siguiente código

```

#include <stdio.h>
int main()
{
    for(double iDb=0.0; iDb<2; iDb+=0.1)
        printf("%.15f \n", iDb);

    printf("\n");
    for(float iFt=0.0; iFt<2; iFt+=0.1)
        printf("%.15f \n", iFt);

    printf("\n");
    return 0;
}

```

Compare las salidas de cada lazo *for* y justifique las diferencias.

34. El siguiente código es casi idéntico al del problema anterior, sin embargo su comportamiento es diferente. Explique el motivo.

```
#include <stdio.h>
int main()
{
    for(double iDb=0.0; iDb<3; iDb+=0.25)
        printf("%.15lf \n", iDb);

    printf("\n");
    for(float iFt=0.0; iFt<3; iFt+=0.25)
        printf("%.15lf \n", iFt);

    printf("\n");
    return 0;
}
```