

Computación 2023



Clase Anterior

- Hardware (cpu, memorias, periféricos, USB)
- Software

computacion.fcaglp.unlp.edu.ar

Software

- Sistemas Operativos (SO)
- Los SO pueden ser multitarea (multitasking) y/o multiusuario (multiuser)
- Para dispositivos se lo llama “Firmware”.
- Se utiliza un lenguaje humano para programar las órdenes, no se suele hacer en lenguaje de la CPU.
- Los lenguajes son Compilados o Interpretados. Aunque hay formas modernas que son mezclas de ambos tipos.

Lenguajes

- Fortran —> Lenguaje compilado
- Python —> Interpretado (pero*...)

* Se llaman en forma interpretada (frontend) a rutinas ya compiladas (backend)...

Hay muchos lenguajes. Su uso depende para que se quiera, cada uno se lo desarrolló con una meta en particular. Pero así y todo, todos tienen estructuras de programación similares.

Lenguaje compilado:

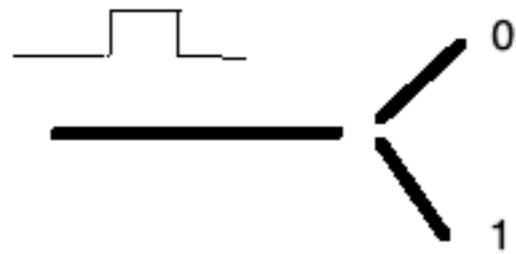
Creo el programa en lenguaje Fortran (humano)

Lo compilo y creo otro programa que es el ejecutable (con órdenes binarias de la cpu).

Corro el ejecutable.

Números Binarios

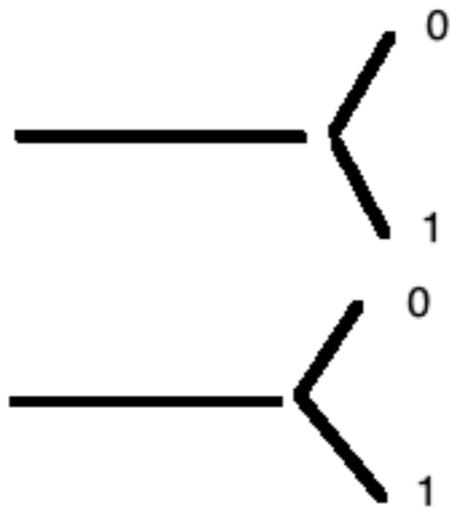
Un cable



0,1

Hay 2 combinaciones

Dos cables



0 0
0 1
1 0
1 1

Hay 4 combinaciones

Base Binaria

Cantidad de Cables N (bits)	Cantidad de números que puedo escribir	Potencia de 2 (2^n)
1	2	1 bit $\rightarrow 2^1 = 2$
2	4	2 bits $\rightarrow 2^2 = 4$
3	8	3 bits $\rightarrow 2^3 = 8$
4	16	4 bits $\rightarrow 2^4 = 16$
...
8	256	8 bits $\rightarrow 2^8 = 256 \leftarrow$ Byte
...
10	1024	10 bits $\rightarrow 2^{10} = 1024 \leftarrow$ KiloByte [KB]
...
20	1048576	20 bits $\rightarrow 2^{20} = 1048576 \leftarrow$ MegaByte [MB]
...
N	M	N bits $\rightarrow 2^N = M$

Tabla 2.1. relación entre bits, Bytes y números binarios

Bits & Bytes

- A un grupo de 8 bits (un 0 o un 1) se lo llama Byte
- El Byte es la unidad con que se define la información en los sistemas digitales
- Hay una confusión constante con las medidas MKS. En binario $1\text{KByte} = 1024 \text{ Bytes} = 2^{10} \text{ Bytes}$
- También lo hay entre bits y Bytes
- Estas confusiones son aprovechadas por los fabricantes de equipos informáticos para parecer que venden mejores dispositivos...(en fin...)

Nomenclatura de Unidades

Símbolo	Prefijo	MKS	Binario	Diferencia Porcentual	Ref
K	kilo	$10^3 = 1000^1$	$2^{10} = 1024^1$	2.40 %	
M	mega	$10^6 = 1000^2$	$2^{20} = 1024^2$	4.86 %	Memoria Cache
G	giga	$10^9 = 1000^3$	$2^{30} = 1024^3$	7.37 %	Memoria RAM/SSD
T	tera	$10^{12} = 1000^4$	$2^{40} = 1024^4$	9.95 %	Discos Rígidos/SSD
P	peta	$10^{15} = 1000^5$	$2^{50} = 1024^5$	12.59 %	Grandes Servers
E	exa	$10^{18} = 1000^6$	$2^{60} = 1024^6$	15.29 %	Datacenters/Nube
Z	zetta	$10^{21} = 1000^7$	$2^{70} = 1024^7$	18.06 %	
Y	yotta	$10^{24} = 1000^8$	$2^{80} = 1024^8$	20.89 %	

Representación de un número

- La base de un sistema numérico es el primer número que tengo que representar como combinación de los anteriores: En base decimal es 10, en binario el 2 es el 10.
- La potencia de cada cifra de un número cuando lo expresé, ejemplo: El número 45 en base 10 se sobreentiende que es $4 \times 10^1 + 5 \times 10^0 = 45$
- Ese número en binario es: 101101, ya que $1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 32 + 8 + 4 + 1 = 45$
- Con respecto a los dígitos fraccionarios de un número esos se representaron como potencias negativas de la base. En binario sería en potencias de $2^{-1}, 2^{-2}, 2^{-3}, \dots$
- Se usan otras bases para representar números octal (base 8) y Hexadecimal (base 16)

Pero hay otras Bases que se usan, por ejemplo, base 16 (Hexadecimal)

Número Decimal Base 10	Binario Base 2	Hexadecimal Base 16
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Tabla 2.2. En esta tabla se ven la representación de los primeros números en distintas bases. NOTE que las letras A,B,C,D,E y F en el sistema hexadecimal son números

Hexadecimales

- Por convención se les escribe con 0x (o #) delante para avisar que es base 16
- 0x9AD3, 0x45FC
- 45 en base 16 es 0x2D $\rightarrow 2 \times 16^1 + D \times 16^0 = 32 + 13 = 45$
- Un número Hexadecimal ocuparía 4 bits en binario, por lo que dos Hexas son un BYTE
- Esa última propiedad hace que sean los preferidos de los ingenieros, como equivalente a los binarios (ya que estos se vuelven largos muy rápidamente)
- Se usan para indicación de colores en el sistema RGB

Repaso de la clase

- Un bit es un 0 un 1
- 8 bits son 1 Byte. ←— Y esta es la unidad fundamental de medida en las computadoras.
- Hay otra base muy asociada con la ingeniería de las computadoras que es la base 16 (Hexadecimal).
- 1 Hexadecimal—> 4 bits. 2 Hexa—> 1 Byte.

Variables y constantes

- Las computadoras usan números binarios y con ello deben representar los números que nosotros usamos.
- Para ello se usan una cantidad de Bytes (varios!)
- Hay también variables y constantes no numéricas (¿?)
- Los números son enteros, reales y complejos. Pero también hay Booleanos y textos que se pueden procesar en un programa

Enteros

(incluye los naturales)

- Se usa una cantidad de Bytes para que los bits de estos sean potencias binarias.
- Pueden ser enteros de 2,4 u 8 Bytes.
- La CPU tiene una unidad propia para trabajar con enteros solamente.
- Puedo utilizar un bit para que el número tenga signo (+ o -)

Bytes n	bits $8n$	Sin signo max. número ($2^n - 1$)	con signo intervalo de números
1	8	256	(-128,127)
2	16	65535	(-32768,32767)
4	32	4294967295	(-2147483648,2147483647)
8	64	18446744073709551615	(-9223372036854775808,9223372036854775807)

Reales (Flotantes)

- Se usan 4,8 o a veces 16 Bytes
- Los bits se distribuyen en mantisa y exponente
- \pm mantisa $\times 2^{\pm \text{exponente}}$
- La Norma IEEE-754 usa 8 bits para el exponente y 24 para la mantisa y el signo. El exponente también tiene signo
- En el caso de 8 Bytes se lo considera Doble Precisión o real^*8 . El número normal tiene 4 Bytes (real^*4)

- Se usa la notación anglosajona “.” y “,” se usan al revés de lo que nosotros estamos acostumbrados.
- La notación científica se escribe con la letra E. 1.23×10^{-8} se escribe como 1.23E-8
- Complejos -> son dos reales, uno para la parte real y otro para la imaginaria
- Lógicas: Solo guardan un verdadero o un falso. Usan 1 Byte, pero de los 8 bits sólo se necesita 1 bit (desperdician 7 bits)
- Caracteres: Usan un Byte por letra (o más según la norma Ascii/Unicode).