

Semana 02

Datos e información

Datos

- Sugiere hechos, cifras en bruto, se expresa en símbolos, números o letras.
- No tienen organización.
- Requieren proceso para significado.
- Se interpretan para significado.

Información

- Está delimitada por los datos.
- Es conocimiento que se remodela y clasifica.
- Se nutre de los datos.
- Se obtiene de distintas fuentes.

Diferencias

Datos	Información
Son variables que ayudan al desarrollo de ideas/conclusiones.	Son datos con significado.
Son textos y valores numéricos.	Se refina de los datos actuales.
No dependen de la información.	Dependen de los datos.
Bits y bytes son la unidad de medida.	Se mide en unidades de significado como tiempo, cantidad, etc.
Se puede estructurar: tablas, gráficas, árboles.	Se estructura como: lenguaje, ideas, pensamientos.
No tienen un propósito específico.	El significado se asigna con base en la interpretación.
Es conocimiento a bajo nivel.	Es un segundo nivel de conocimiento.
No ayuda directamente en la toma de decisiones.	Ayuda directamente en la toma de decisiones.
Colección de hechos, por sí mismos carecen de significado.	Pone a los hechos en contexto.

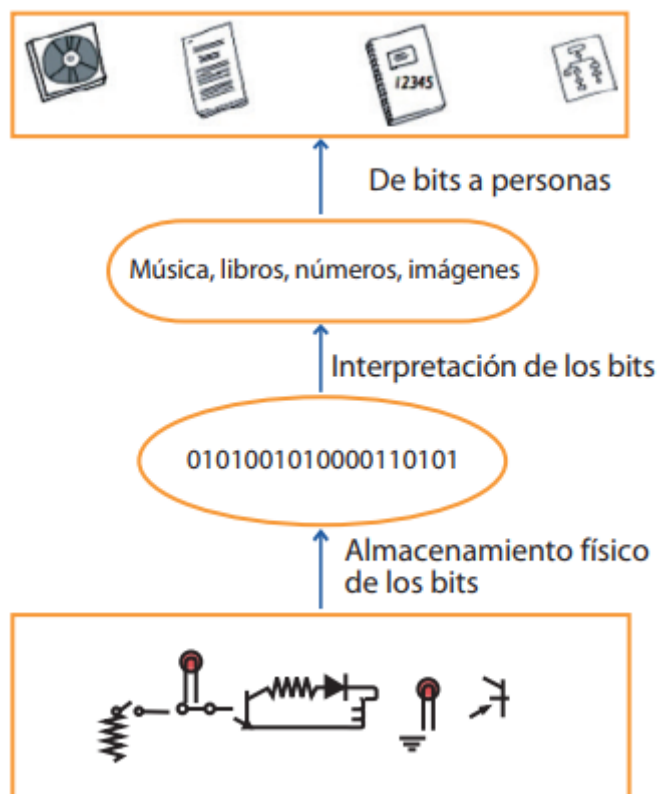
Ejemplo: puntaje de examen de un estudiante.

Ejemplo: promedio de la clase para determinar aprovechamiento.

Representación

- Sonidos de campana en la antigüedad.
- Encendido/Apagado. (Bits 0, 1)
- Manera simple de representación de la información.

Longitud $\log_2 x = n$ significa que $2^n = x$



Representación de números

Decimal

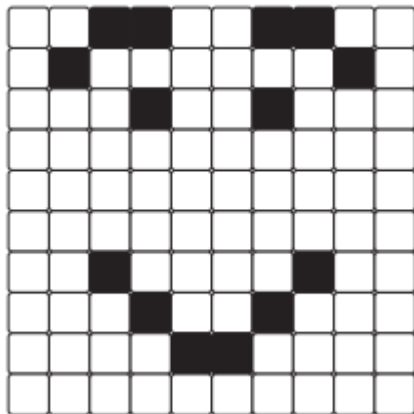
$$4649 = 4 \times 1000 + 6 \times 100 + 4 \times 10 + 9 = 4 \times 10^3 + 6 \times 10^2 + 4 \times 10^1 + 9 \times 10^0$$

Binario

$$\begin{aligned}
 1011_2 &= 1 \times 2^3 + 0 \times 2^2 \\
 &+ 1 \times 2^1 + 1 \times 2^0 = 1 \times 8 \\
 &+ 1 \times 4 + 1 \times 2 + 1 \times 1 = 8 \\
 &+ 0 + 2 + 1 = 11_{10}
 \end{aligned}$$

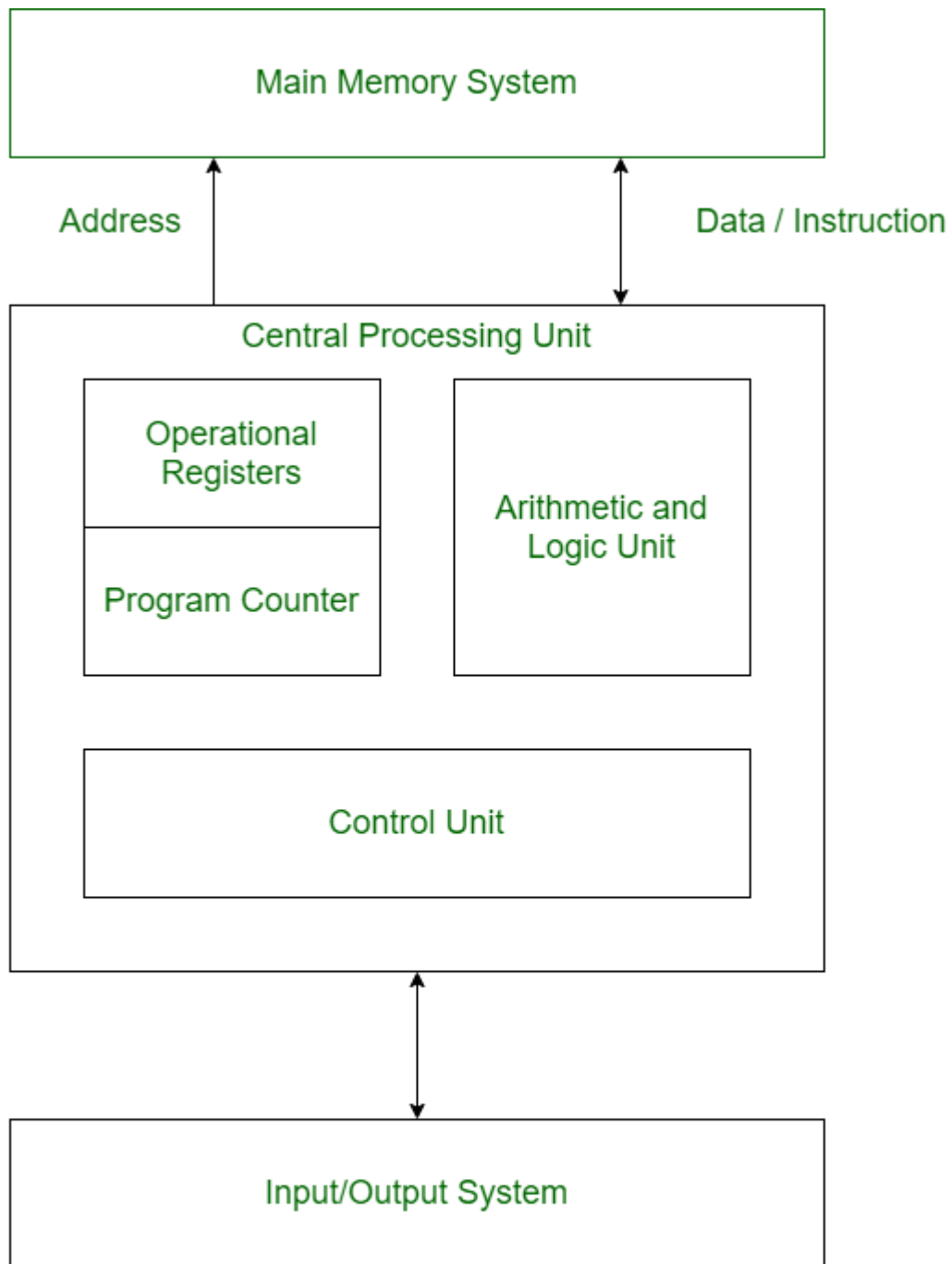
Número	$8 = 2^3$	$4 = 2^2$	$2 = 2^1$	$1 = 2^0$
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Representación semi-tangible

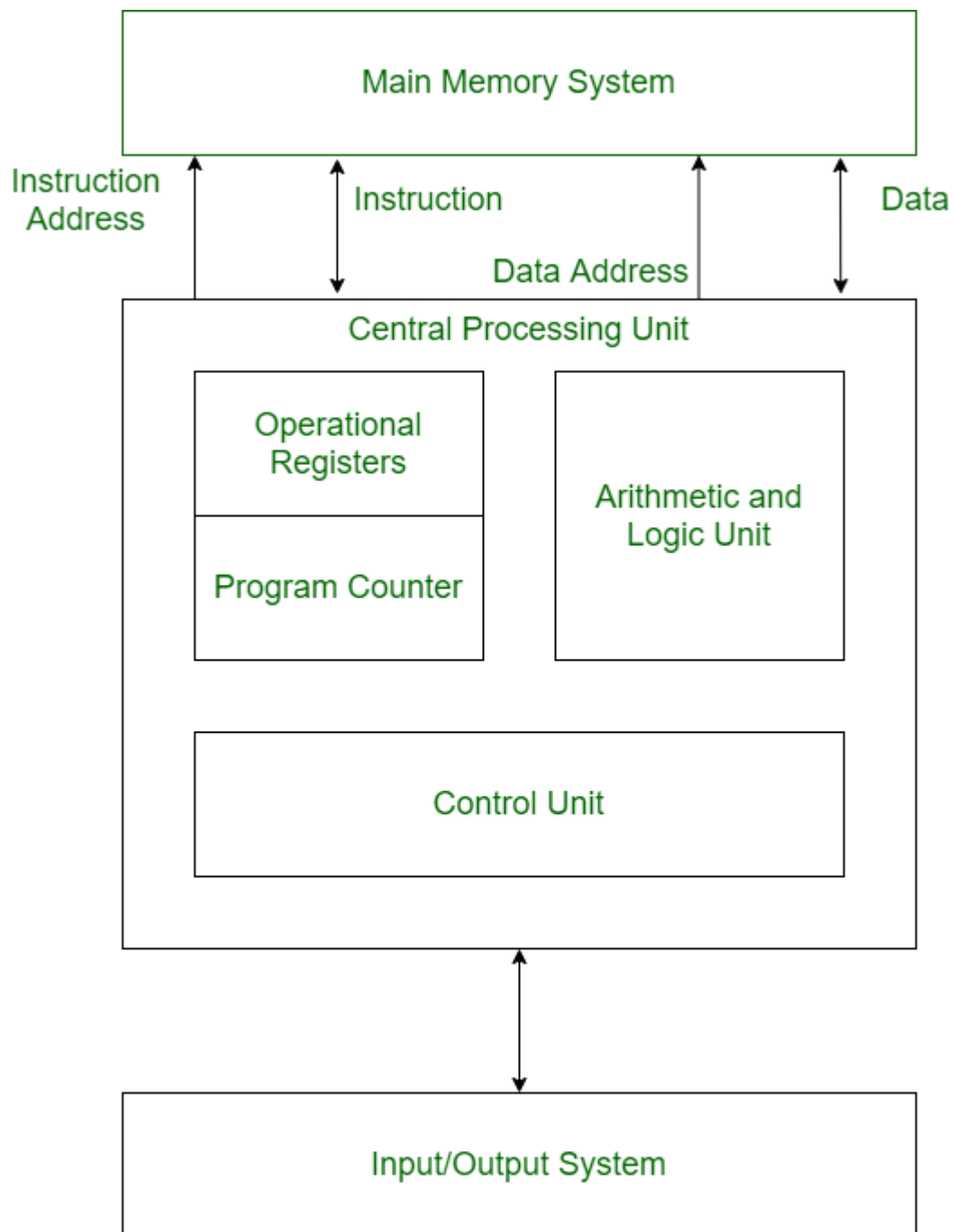


0011001100
0100000010
0001001000
0000000000
0000000000
0000000000
0010000100
0001001000
0000110000
0000000000

Arquitecturas de Computación Actuales



Von Neumann Architecture



Harvard Architecture

Arquitectura Von Neumann	Arquitectura Harvard
Arquitectura basada en almacenar programas de cómputo.	Basado en Harvard Mark I

Direccionamiento físico de memoria compartida para instrucciones y datos.	Direccionamiento físico de memoria separado para instrucciones y datos.
Bus en común para transferencia de datos e instrucciones.	Buses separados para transferencia de datos e instrucciones.
Dos ciclos de reloj requieren para ejecutar una instrucción.	Una instrucción se ejecuta en un ciclo.
Más barato computacionalmente.	Más costoso computacionalmente.
CPU no puede acceder a instrucciones y leer/escribir al mismo tiempo.	CPU puede acceder a instrucciones y leer/escribir al mismo tiempo.
PCs.	Microcontroladores y DSP.

Paradigmas de programación

Paradigma de programación Imperativa - ¿Cómo hacer?

- Basada en arquitectura Von Neumann.
- Funciona cambiando el estado del programa mediante sentencias de asignación.
- Hace tareas paso a paso cambiando de estado.
- Su enfoque principal está en la forma de lograr el objetivo.
- Consta de varias declaraciones y después de la ejecución de todo se almacena el resultado.

Ventajas

- Implementación simple.
- Contiene bloques de control, variables.

Desventajas

- Problemas complejos.
- No permite programación paralela.
- Menos eficientes, menos productivos.

Programación estructurada

- C
- C++
- Pascal

Programación Orientada a Objetos

- Colección de Clases y Objetos diseñados para comunicación entre elementos.
- Mayor énfasis en datos, menor énfasis en procedimientos.
- Mayor manejo de problemas de la vida real.
- Ventajas adicionales: Seguridad en los datos, Herencia, Reusabilidad de código, Flexibilidad y abstracción.
- Java, C++, .NET, Python, Ruby.

Paradigma de programación declarativa - ¿Qué hacer?

- Lógica, Funcional y de Base de Datos.
- Expresa la lógica de la computación sin hablar del flujo de control.
- Se considera a los programas como teorías de alguna lógica.
- Se declara el resultado que deseamos, en lugar de como se produce.