

95:10 MODELACIÓN NUMÉRICA
INTRODUCCIÓN A LA COMPUTACIÓN

Producido por: Angel Menéndez

Fecha: 09-jun-12

INDICE

1. SIMULACIÓN DE PROCESOS

2. MODELOS MATEMÁTICOS

2.1. Concepto

2.2. Modelos Empíricos

2.3. Modelos de Procesos

3. TEORÍA GENERAL DE SISTEMAS

3.1. Concepto

3.2. Sistemas abiertos y cerrados

3.3. Ejemplo

4. SIMULACIÓN NUMÉRICA

4.1. Concepto

4.2. Ejemplo

5. ORGANIZACIÓN FÍSICA DE LA COMPUTADORA (HARDWARE)

5.1. Dispositivos periféricos

5.2. Memoria

5.3. CPU

6. PROGRAMAS (SOFTWARE)

6.1. Tipo de software

6.2. Lenguajes de programación

6.3. Programación

7. BASES DE DATOS

7.1. Concepto

7.2. DBMS

BIBLIOGRAFÍA

1. SIMULACIÓN DE PROCESOS

La simulación de procesos significa representar un proceso mediante otro que comparte las principales características, pero que se encuentra bajo control del usuario.

La simulación abarca desde cuestiones tan elementales como simular una toma de examen, hasta el entrenamiento virtual de pilotos mediante simuladores de navegación ó de cirujanos para llevar adelante intervenciones quirúrgicas. Muy difundidas son las aplicaciones recreativas en la computadora, las cuales crean ambientes artificiales, con sonido incluido, que logran una representación muy verosímil de cualquier tipo de contenido. Otra aplicación interesante es en el campo de la asistencia para mitigar minusvalías físicas. También existe la simulación virtual de la composición musical, que, al generar una interfaz entre el usuario y la aplicación, evita que la habilidad en el manejo de un instrumento se convierta en un obstáculo.

2. MODELOS MATEMÁTICOS

2.1. Concepto

Muchas de las simulaciones de procesos se basan en modelos matemáticos. Un modelo matemático es una representación matemática de un fenómeno del mundo real. El objetivo del modelo matemático es entender el fenómeno (es decir, poder explicarlo en términos de procesos elementales) y, en la medida de lo posible, predecir su comportamiento futuro.

El proceso para elaborar un modelo matemático es el siguiente:

- I. Identificar claramente el problema.
- II. Formular el modelo matemático del problema, identificando variables (independientes y dependientes) y estableciendo las hipótesis de base que le dan sustento.
- III. Resolver el problema matemático mediante las técnicas más apropiadas.
- IV. Comparar los resultados obtenidos con datos reales. Si existen diferencias inaceptables, ajustar el modelo.

Los modelos matemáticos pueden ser empíricos o de procesos. Los modelos empíricos consisten en establecer relaciones funcionales entre variables dependientes e independientes en base a la observación de las tendencias de esas relaciones que muestran los datos (por ejemplo, relación entre el espacio recorrido y el tiempo invertido en recorrerlo), y ajustar los parámetros de modo de lograr el mejor acuerdo posible. Los modelos de procesos, en cambio, se establecen como ecuaciones (algebraicas y/o diferenciales) basadas sólo en principios (por ejemplo, principio de conservación de la masa), con valores relativamente universales de los parámetros. Algunos modelos pueden ser una combinación de submodelos de procesos y empíricos.

2.2. Modelos Empíricos

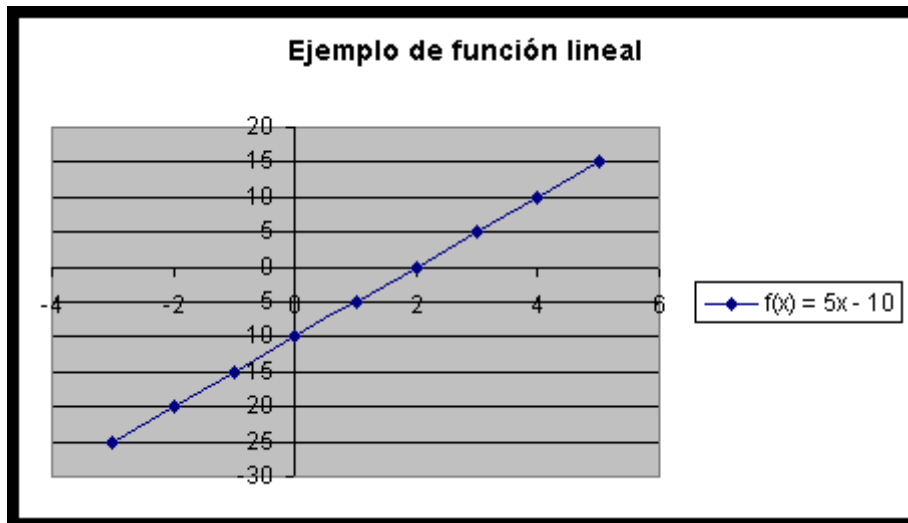
A continuación se describen relaciones funcionales típicas que se utilizan en modelos empíricos.

Función Lineal

Se dice que una función es lineal cuando su gráfica es una línea recta; tiene la forma:

$$y = f(x) = mx + b$$

donde m representa la pendiente de la recta y b la ordenada al origen. Este tipo de funciones representa un crecimiento a tasa constante. Su dominio e imagen son todos los números reales.



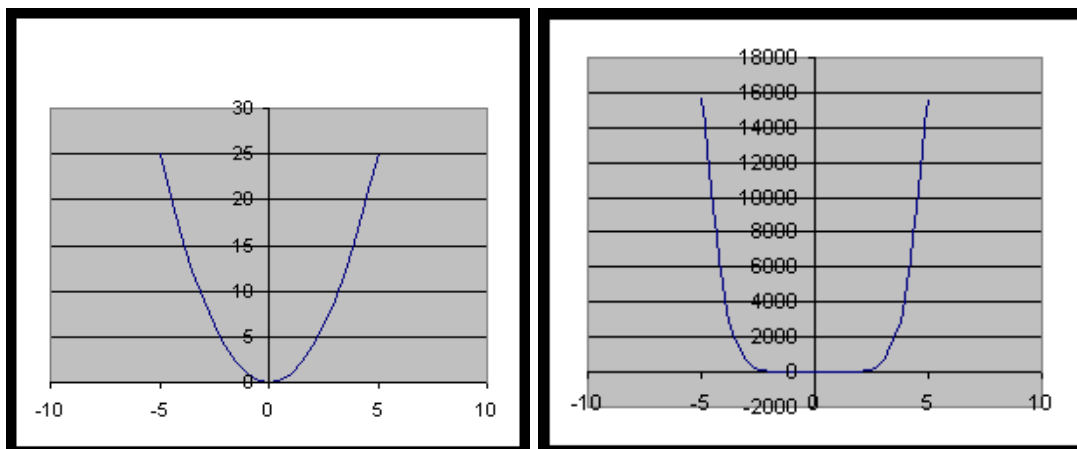
Funciones Polinómicas

Una función polinómica tiene la forma siguiente:

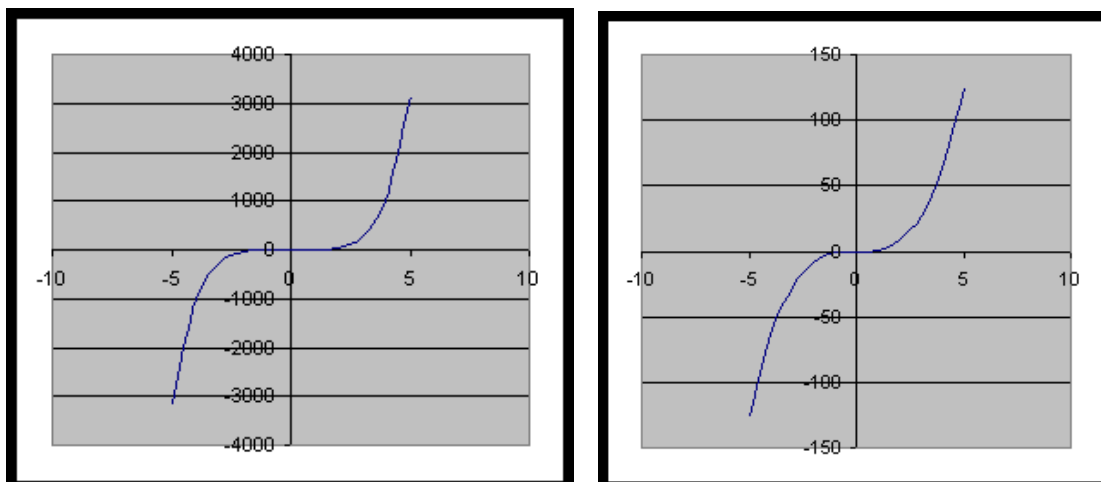
$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0$$

donde n es un entero positivo y $a_0, a_1, a_2, \dots, a_n$ son constantes llamadas coeficientes del polinomio. El dominio de todos los polinomios son todos los números reales $(-\infty, \infty)$.

El grado del polinomio es la potencia del primer término, n . Los polinomios de grado uno son las funciones lineales. Los polinomios de segundo grado son las funciones cuadráticas; su gráfica es una parábola. El polinomio de tercer grado es llamada función cúbica.



Las dos gráficas anteriores son ejemplos de funciones pares: x^2 y x^6



Las dos gráficas anteriores son ejemplos de funciones impares: x^3 y x^5

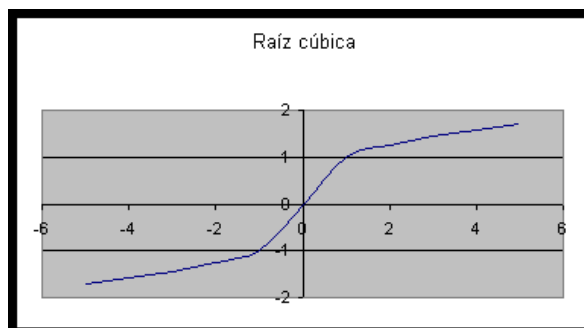
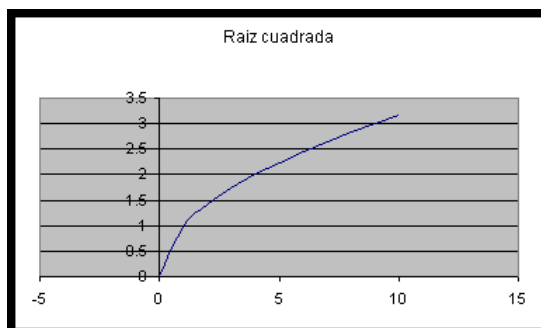
Funciones Potencia

Una función es llamada potencia cuando tiene la forma:

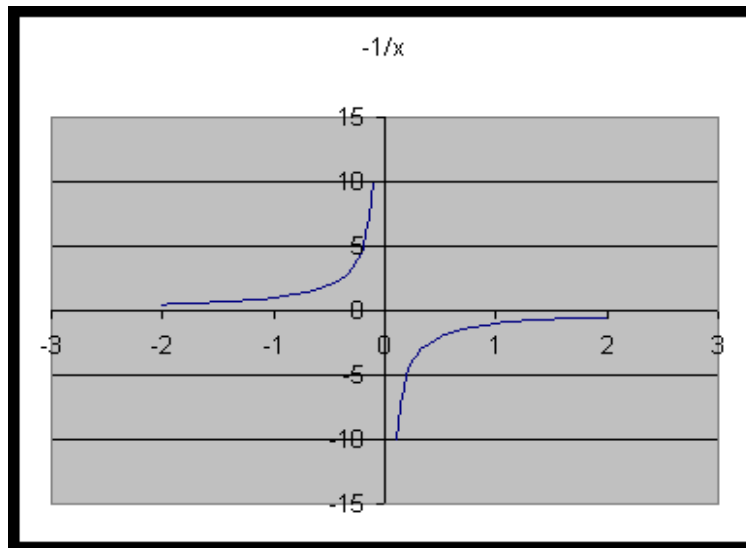
$$f(x) = x^a$$

donde a es una constante. Hay varios casos:

- $a = n$, n entero positivo. La forma general de la gráfica depende de si n es par o impar; si n es par, la gráfica de f es similar a la parábola $y = x^2$; si es impar, la gráfica se parece a la función $y = x^3$. En cualquier caso, cuando n crece la gráfica se vuelve más plana cerca de 0 y más empinada cuando $|x|$ es mayor o igual a 1.
- $a = 1/n$, n entero positivo. Es una función raíz. Al igual que en el caso anterior, su gráfica depende de la paridad de n ; si n es par su gráfica es similar al de la raíz cuadrada; si n es impar su gráfica será similar al de la raíz cúbica.



- $a = -1$. Esta función es llamada recíproca. Su gráfica corresponde a una hipérbola cuyas asíntotas son los ejes de coordenados.



Funciones Racionales

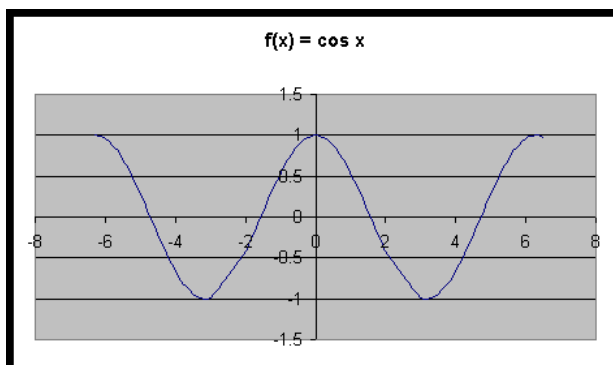
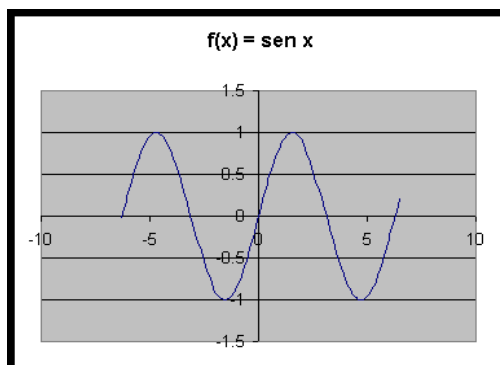
Una función es llamada racional cuando es una razón o división de dos polinomios:

$$f(x) = P(x) / Q(x)$$

Su dominio lo constituyen todos los valores de x que no hagan que $Q(x) = 0$.

Funciones Trigonométricas

El argumento de estas funciones se idéntica con ángulos, medidos en radianes. En el caso específico del seno y el coseno, su dominio es $(-\infty, \infty)$ y su imagen $[-1, 1]$.

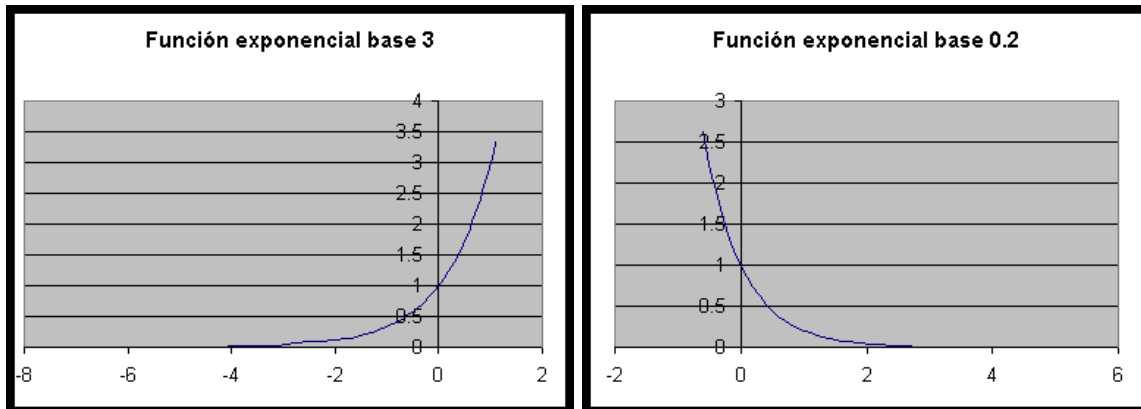


Funciones Exponenciales

Se llama función exponencial a aquella de la forma

$$f(x) = a^x$$

donde la base a es una constante positiva. Su dominio es $(-\infty, \infty)$ y su imagen $(0, \infty)$. Si la base de la función exponencial es mayor a 1 la gráfica es ascendente; si la base es menor a 1 la gráfica es descendente.

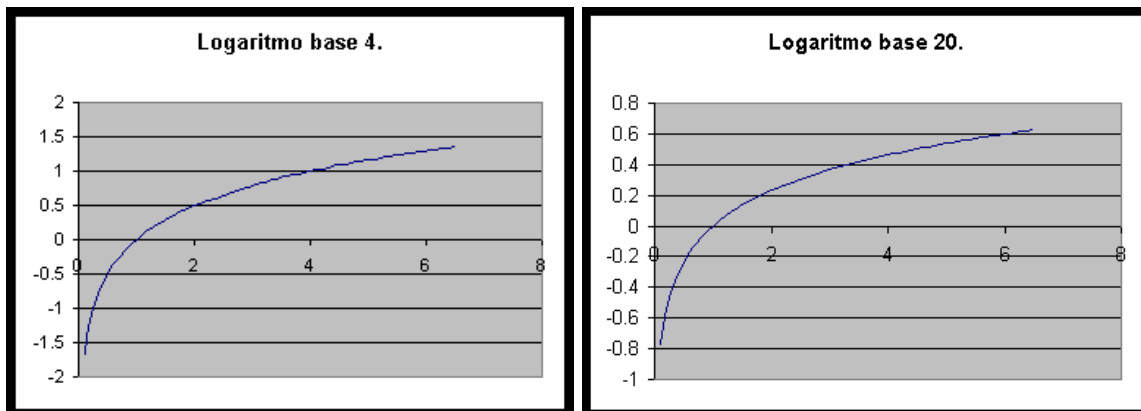


Funciones Logarítmicas

Son funciones que tienen la forma

$$f(x) = \log_a x$$

donde la base a es una constante positiva. Son las funciones inversas a las exponenciales; por lo tanto su dominio es $(0, \infty)$ y su imagen $(-\infty, \infty)$. A medida que la base del logaritmo es mayor, la gráfica se pega más al eje y .



Funciones Trascendentes

Engloba a todas aquellas funciones que no son algebraicas (estas últimas son las que involucran sólo adición, sustracción, división y multiplicación de variables). Las funciones trascendentes incluye a las trigonométricas, logarítmicas, exponenciales y trigonométricas inversas, entre otras.

2.3. Modelos de Procesos

Un ejemplo simple de modelo de procesos es el de decaimiento de la concentración de una sustancia con el tiempo, $c(t)$, por transformación mediante una reacción de primer orden

$$\frac{dc}{dt} = -kt$$

donde k es el coeficiente de reacción (esta es parte de un submodelo empírico de decaimiento). El modelo se completa con la condición inicial $c(0)=c_0$ (valor conocido).

Si k es constante, este problema diferencial tiene la solución cerrada

$$c(t) = c_0 e^{-kt}$$

En cambio, si k es una función de c , la ecuación tiene que ser resuelta por métodos numéricos.

3. TEORÍA GENERAL DE SISTEMAS

3.1. Concepto

La Teoría General de Sistemas (TGS) apunta a producir teorías y formulaciones conceptuales que pueden conducir a aplicaciones. Es decir, no busca en sí misma solucionar problemas o intentar soluciones prácticas.

Existe una tendencia hacia la integración de diversas ciencias naturales y sociales a través de una teoría de sistemas. Dicha teoría es una manera más amplia de estudiar los campos no-físicos del conocimiento científico, especialmente los asociados a las ciencias sociales. Al desarrollar principios unificadores, que atraviesan verticalmente los universos particulares de las diversas ciencias involucradas, la teoría de sistemas produce una aproximación al objetivo de la unidad de la ciencia.

La TGS afirma que las propiedades de los sistemas no pueden ser descritas en términos de sus elementos separados, sino que su comprensión se logra cuando se estudian globalmente. La TGS se fundamenta en tres premisas básicas:

- I. Cada sistema existe dentro de otro sistema más grande.
- II. Los sistemas son abiertos. Cada sistema que se examine, excepto el menor y el mayor, recibe y descarga algo en los otros sistemas, generalmente en los contiguos. Los sistemas abiertos se caracterizan por un proceso de cambio infinito con su entorno, que son los otros sistemas. Cuando el intercambio cesa, el sistema se desintegra, esto es, pierde sus fuentes de energía.
- III. Las funciones de un sistema dependen de su estructura. Para los sistemas biológicos y mecánicos esta afirmación es intuitiva. Los tejidos musculares por ejemplo, se contraen porque están constituidos por una estructura celular que permite contracciones.

El análisis de las organizaciones vivas revela "lo general en lo particular" y muestra las propiedades generales de las especies, que son capaces de adaptarse y sobrevivir en un ambiente típico. Los sistemas vivos, sean individuos u organizaciones, deben ser analizados como sistemas abiertos que mantienen un continuo intercambio de materia/energía/información con el ambiente. Las teorías tradicionales han visto a la organización humana como un sistema cerrado. Eso ha llevado a no tener en cuenta el ambiente, provocando poco desarrollo y comprensión de la retroalimentación ("feedback"), básica para sobrevivir. La TGS permite reformular los fenómenos dentro de un enfoque global, para integrar asuntos que son, la mayoría de las veces, de naturaleza completamente diferente.

3.2. Sistemas abiertos y cerrados

El sistema abierto es un conjunto de partes en interacción constituyendo un todo sinérgico, orientado hacia determinados propósitos y en permanente relación de interdependencia con el ambiente externo. El sistema abierto es influenciado por el medio ambiente e influye sobre él, alcanzando un equilibrio dinámico. La categoría más importante de los sistemas abiertos son los sistemas vivos.

Existen diferencias entre los sistemas abiertos (como los sistemas biológicos y sociales, a saber, células, plantas, el hombre, la organización, la sociedad) y los sistemas cerrados (como los sistemas físicos, las máquinas, el reloj, el termóstato). Las principales son las siguientes:

- El sistema abierto interactúa constantemente con el ambiente en forma dual, o sea, lo influye y es influenciado. El sistema cerrado no interactúa.
- El sistema abierto puede crecer, cambiar, adaptarse al ambiente y hasta reproducirse bajo ciertas condiciones ambientales. El sistema cerrado no.
- Es propio del sistema abierto competir con otros sistemas, no así el sistema cerrado.

3.3. Ejemplo

Si se aplica la TGS a la administración de una empresa, ésta se concibe como una estructura que se reproduce a través de un sistema de toma de decisiones, tanto individual como colectivamente.

Desde un punto de vista histórico, la evolución de este modelo siguió los siguientes pasos:

- La teoría de la administración científica usó el concepto de sistema hombre-máquina, pero se limitó al nivel de trabajo fabril.
- La teoría de las relaciones humanas amplió el enfoque hombre-máquina a las relaciones entre las personas dentro de la organización. Provocó una profunda revisión de criterios y técnicas gerenciales.
- La teoría estructuralista concibió la empresa como un sistema social, reconociendo que hay tanto un sistema formal como uno informal dentro de un sistema total integrado.
- La teoría del comportamiento aportó la teoría de la decisión, donde la empresa se ve como un sistema de decisiones, ya que todos los participantes de la empresa toman decisiones dentro de una maraña de relaciones de intercambio, que caracterizan al comportamiento organizacional.
- Después de la Segunda Guerra Mundial, a través de la teoría matemática se aplicó la investigación operacional para la resolución de problemas grandes y complejos con muchas variables.
- La teoría de colas fue profundizada y se formularon modelos para situaciones típicas de prestación de servicios, en los que es necesario programar la cantidad óptima de servidores para una esperada afluencia de clientes.

Al igual que los organismos vivos, las empresas tienen seis funciones primarias estrechamente relacionadas entre sí:

- *Ingestión:* Las empresas hacen o compran materiales para ser procesados. Adquieren dinero, máquinas y personas del ambiente para asistir otras funciones, tal como los organismos vivos ingieren alimentos, agua y aire para suplir sus necesidades.
- *Procesamiento:* Los animales ingieren y procesan alimentos para ser transformados en energía y en células orgánicas. En la empresa, la producción es equivalente a este ciclo. Se procesan materiales y se desecha lo que no sirve, habiendo una relación entre las entradas y salidas.

- *Reacción al ambiente:* El animal reacciona a su entorno, adaptándose para sobrevivir, debe huir o si no atacar. La empresa reacciona también, cambiando sus materiales, consumidores, empleados y recursos financieros. Se puede alterar el producto, el proceso o la estructura.
- *Provisión de las partes:* Partes de un organismo vivo pueden ser suplidas con materiales, como la sangre abastece al cuerpo. Los participantes de la empresa pueden ser suplidos con datos de compras, producción, ventas o contabilidad, y se les recompensa bajo la forma de salarios y beneficios. El dinero es muchas veces considerado la sangre de la empresa.
- *Regeneración de partes:* Las partes de un organismo pierden eficiencia, se enferman o mueren y deben ser regeneradas o relocalizadas para sobrevivir en el conjunto. Miembros de una empresa envejecen, se jubilan, se enferman, se desligan o mueren. Las máquinas se vuelven obsoletas. Tanto hombres como máquinas deben ser mantenidos o relocalizados, de ahí la función de personal y de mantenimiento.
- *Organización:* De las funciones, es la requiere un sistema de comunicaciones para el control y toma de decisiones. En el caso de los animales, que exigen cuidados en la adaptación. En la empresa se necesita un sistema nervioso central, donde las funciones de producción, compras, comercialización, recompensas y mantenimiento deben ser coordinadas. En un ambiente de constante cambio, la previsión, el planeamiento, la investigación y el desarrollo son aspectos necesarios para que la administración pueda hacer ajustes.

4. SIMULACIÓN NUMÉRICA

4.1. Concepto

La gran evolución de los métodos informáticos, tanto a nivel de hardware como software, ha permitido afrontar la resolución de complejos sistemas de ecuaciones físico-matemáticas cuya resolución analítica resulta prácticamente imposible. De hecho, muchos de esos problemas ya se encontraban planteados hacía tiempo, faltando sólo un medio adecuado para la obtención de resultados prácticos.

La simulación numérica intenta, entonces, reproducir la realidad a partir de resolución numérica, mediante la computadora, de ecuaciones matemáticas que describen dicha realidad (modelo matemático). Obviamente, la simulación numérica sólo puede ser, entonces, tan precisa como lo es el modelo matemático, lo cual fija su alcance.

4.2. Ejemplo

Para el diseño del nuevo sistema de esclusas del Canal de Panamá (que permitirá el pasaje de buques de mayor tamaño) se utilizó como base la simulación numérica¹. Se empleó como base el sistema de modelación a código abierto OpenFOAM. En la siguiente figura se muestran distribuciones instantáneas de velocidad para distintos instantes del evento de llenado de la cámara de la esclusa desde el sistema hidráulico de llenado/vaciado. Corresponde a un corte horizontal a la altura del plano medio de los puertos de descarga (20 en total). Se observa la capacidad de representar las estructuras turbulentas. Uno de los objetivos principales del diseño fue minimizar las pérdidas energéticas del flujo de agua, de modo de minimizar el tiempo de llenado, apuntando a contribuir al aumento de la frecuencia del pasaje de buques.

¹ http://laboratorios.fi.uba.ar/lmm/canal_panama2.htm

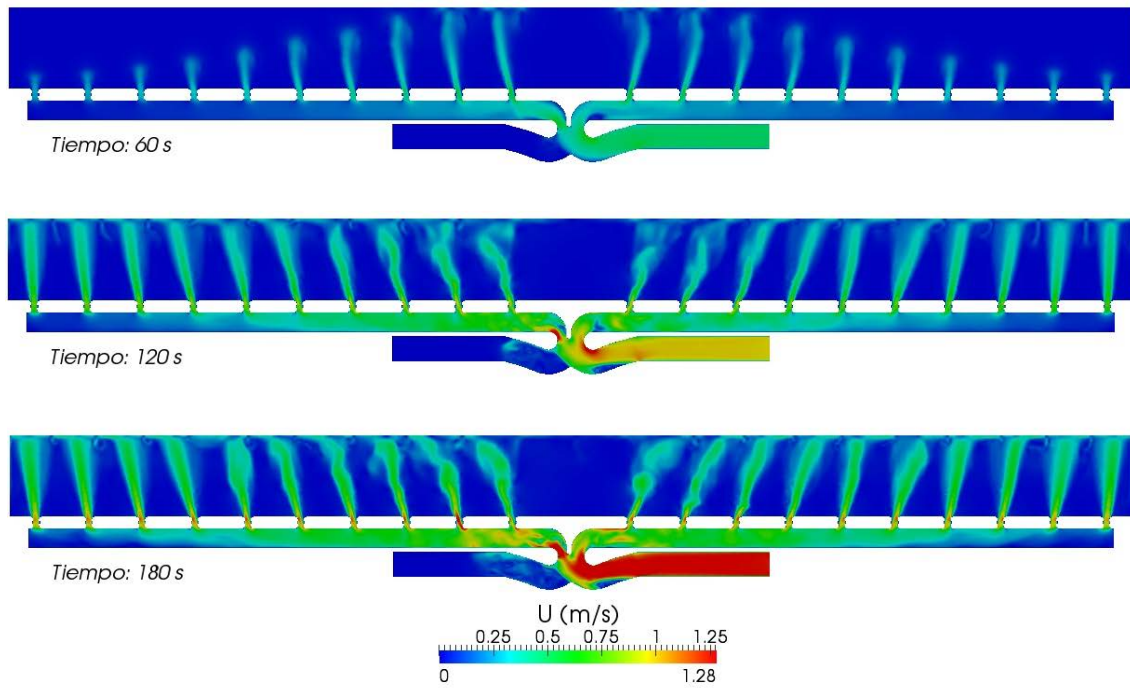


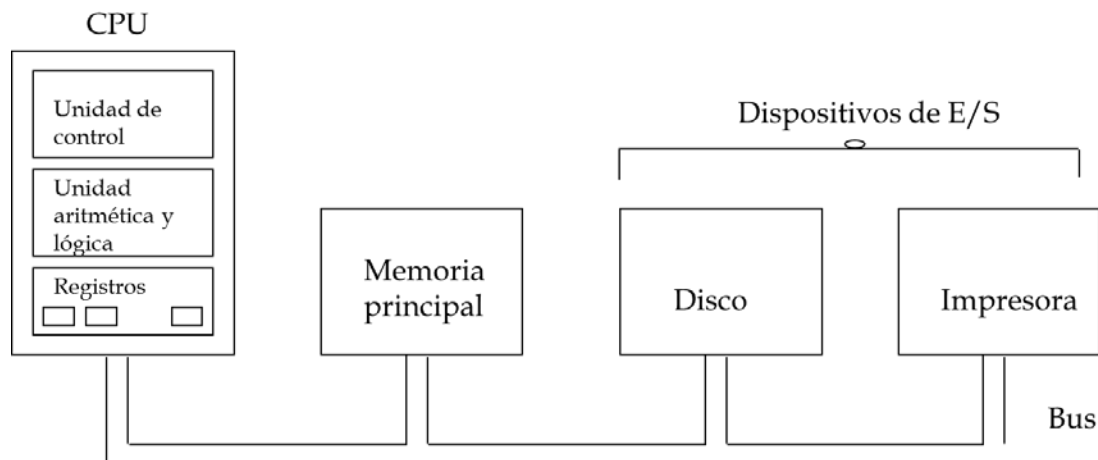
Figura 2.4.3. Distribución del módulo de la velocidad en el plano horizontal a media altura de los puertos, para diferentes pasos de tiempo; a escala de modelo físico y en unidades de modelo físico.

5. ORGANIZACIÓN FÍSICA DE LA COMPUTADORA (HARDWARE)

La computadora puede definirse como un dispositivo mecánico-electrónico que procesa información numérica y alfanumérica.

Físicamente, la computadora puede descomponerse en los siguientes componentes:

- Dispositivos periféricos
- Memoria
- Unidad Central de Procesamiento (CPU)



Estos se describen en lo que sigue.

5.1. Dispositivos periféricos

Los dispositivos periféricos son los que permiten la entrada y salida de información. Por ejemplo:

- Teclado
- Mouse
- Impresora
- Auricular

5.2. Memoria

La memoria es donde se almacena la información. La información puede estar constituida por datos ó instrucciones.

La memoria es una zona de almacenamiento organizada en miles de millones de unidades de almacenamiento individual, denominadas celdas de memoria. La unidad elemental de memoria se llama byte. Está constituida por 8 bits. Un bit almacena un dígito binario (0 ó 1).

Un dato alfanumérico se almacena en un byte. Los datos numéricos pueden ocupar varios (2, 4, 8) bytes consecutivos, según sea su tipo (real simple precisión, real doble precisión, entero, entero largo), constituyendo palabras.

Un byte tiene asociada una dirección, que permite su acceso para almacenamiento ó recuperación. La dirección es única y permanente.

Siempre que se almacena nueva información, esta reemplaza (sobrescribe) a la que existía inicialmente, que es borrada. El contenido puede cambiar mientras se ejecuta un programa.

La capacidad de almacenamiento de la memoria se mide en bytes. 1 Kbyte equivale a 1024 bytes. 1 Mbyte equivale a 1024^2 bytes. 1 Gbytes equivale a 1024^3 bytes.

La información se organiza en unidades independientes que se denominan archivos (ó ficheros). Los archivos pueden ser de programas, de texto, de imágenes. Un archivo se designa por un nombre y una extensión.

Se distingue dos tipos de memoria:

- *Memoria Central, Interna ó Caché:* Es de acceso muy rápido. Pero es volátil y desaparece al desconectar la fuente de energía o de potencia de la computadora.
- *Memoria Auxiliar ó Externa:* Es de acceso más lento. Pero es permanente. Se materializa en: disco duro, pendrive, CD-ROM, DVD, Blue Ray.

5.3. CPU

La CPU tiene las siguientes funciones:

- Dirigir y controlar el proceso de información.
- Recuperar información de la memoria.
- Procesar y manipular la información recuperada.
- Almacenar los resultados de los procesos en memoria.

Se distinguen dos componentes:

- *Unidad de Control:* Coordina las actividades. Determina, ordena y sincroniza los procesos que realiza la computadora.
- *Unidad Aritmético-lógica:* Realiza las operaciones aritméticas y lógicas.
- *Registros:* Pequeña memoria de alta velocidad para almacenar resultados intermedios e información de control.

6. PROGRAMAS (SOFTWARE)

6.1. Tipo de software

Un programa ó software es una lista de instrucciones que indican las operaciones que debe realizar el hardware.

Se distingue dos tipos de software:

- Software del sistema
- Software de aplicaciones

El software del sistema es el conjunto de programas indispensables para que el sistema funcione. Está constituido por el sistema operativo, los traductores (compiladores e intérpretes de lenguajes de programación) y los utilitarios.

Las principales funciones del sistema operativo son las siguientes:

- Facilitar la escritura y uso de sus propios programas.
- Dirigir las operaciones globales de la computadora.
- Instruir a la computadora para ejecutar otros programas.
- Controlar el almacenamiento y recuperación de archivos (programas y datos) desde la memoria.
- Posibilitar que se pueda introducir y grabar nuevos programas e instruir a la computadora para que los ejecute.

Los programas se escriben en lenguajes de programación (ver más abajo) que permiten la comunicación usuario/máquina. Los programas traductores (compiladores e intérpretes) convierten las instrucciones escritas en lenguajes de programación en instrucciones escritas en lenguaje de máquina, que esta puede entender.

Los utilitarios son los programas que facilitan el uso de las computadoras, proveyendo aplicaciones específicas para usos generales. Por ejemplo, los editores de texto (como el Word), las planillas de cálculo (como el Excel), los armadores de presentaciones (como el Power Point).

Los programas de aplicación es software desarrollado para tareas específicas de cierta especialidad (por ejemplo, para diseño ingenieril, para administración de empresas, etc.).

6.2. Lenguajes de programación

Los lenguajes son un conjunto de códigos que se utilizan para escribir los programas de computadora.

El *lenguaje de máquina* es aquel cuyas instrucciones son directamente entendibles por la computadora y no necesitan traducción posterior para que la CPU pueda comprenderlas y ejecutarlas. Las instrucciones en lenguaje de máquina se expresan en términos de la unidad de memoria más pequeña (bit). En esencia, una secuencia de bits especifica la operación y las celdas de memoria implicadas en una operación.

Las instrucciones en lenguaje de máquina son fáciles de leer por una computadora, pero difíciles para que las manipule un programador. Por esta razón surgieron los lenguajes de programación, más fáciles para escribir y ser leídos por los programadores.

Los lenguajes de programación pueden clasificarse en los de bajo nivel y los de alto nivel.

Los *lenguajes de programación de bajo nivel* manejan instrucciones en códigos nemotécnicos (por ejemplo, ADD = suma, SUB= resta, MPY = multiplicar, DIV=dividir, LDA= cargar acumulador, STO= almacenar). Las palabras nemotécnicas son mas fáciles de manipular que las secuencias de bits. Después que un programa ha sido escrito en un lenguaje de bajo nivel se necesita operar un programa llamado ensamblador, que lo traduce a código de máquina. Los lenguajes de bajo nivel dependen de la máquina.

Los *lenguajes de programación de alto nivel* son aquellos en los cuales las instrucciones o sentencias se escriben con palabras similares a los lenguajes humanos (en general, en inglés), lo cual facilita la escritura y la comprensión por parte del programador. Las características principales de los lenguajes de programación de alto nivel son las siguientes:

- *Transportabilidad:* Un programa escrito en un lenguaje de alto nivel se puede escribir con poca o ninguna modificación en distintos tipos de computadora.
- *Independencia:* Los lenguajes son independientes de la máquina, o sea, una sentencia no depende del diseño de hardware de una computadora en particular.

Los programas escritos en lenguaje de alto nivel (programas fuente) no son entendibles directamente por la máquina. Necesitan ser traducidos a instrucciones en lenguaje de máquina mediante un programa interpretador o uno compilador, según sea el lenguaje. Los programas interpretadores traducen y ejecutan una sentencia fuente cada vez. En cambio, los programas compiladores traducen el programa fuente en un programa objeto, el cual puede ser ejecutado indefinidamente. La ejecución con programas interpretadores hace, en general, mucho más lenta la ejecución; como compensación, con los intérpretes es más fáciles la depuración de errores (debugging).

El BASIC ha sido un típico caso de lenguaje que utilizaba un intérprete. El Quick Basic, más moderno, utiliza un compilador. Fortran, Pascal, C, C++ usan programas compiladores.

6.3. Programación

La programación se basa en la construcción de algoritmos. En su acepción más general, un algoritmo es una secuencia finita de instrucciones de significado preciso, ejecutada por una entidad.

El siguiente es un ejemplo de algoritmo para una actividad de la vida diaria, cuando se quiere preparar café:

Ingredientes
Cafè
Agua
Taza
Azúcar
Cuchara
Preparación:
Verter el agua en la taza
Echar el cafè a la taza con agua
Mover con la cuchara para disolver el cafè
Echar dos cucharadas de azúcar a la taza
Mover con la cuchara para disolver el azúcar
Fin

Los ingredientes son los datos de entrada del algoritmo (las variables que utilizará) y la preparación es el algoritmo.

Los algoritmos deben poseer las siguientes características:

- Deben especificarse los datos de entrada (números, textos) en forma precisa.
- Las instrucciones deben ser precisas (no ambiguas), con lo cual cuando se ejecuta con los mismos datos de entrada debe arrojar los mismos resultados.
- Debe ser correcto, conduciendo entonces a la solución esperada.
- Debe concluir luego de una cantidad finita de pasos.

Por ejemplo, el siguiente algoritmo, que cuenta los números naturales empezando en 1, no terminaría nunca; la máquina entraría en un “loop”:

```

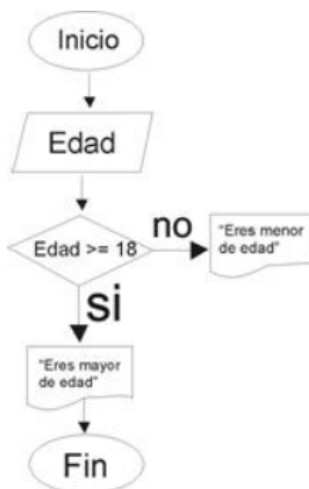
Algoritmo contador_numerico
Variable
    Numero: entero “Al declarar variables no se colocan tildes”
Inicio
    Numero <- 1
    Mientras Numero > 0 hacer
        Imprimir Numero
        Numero <- Numero + 1
    Finmientras
Fin

```

Una forma clásica de expresar un algoritmo es a través de diagramas de flujos. El diagrama de flujo es una representación gráfica del algoritmo que se construye con los siguientes componentes:

Representación	Explicación
	Símbolo utilizado para marcar el inicio y el fin del diagrama de flujo
	Símbolo utilizado para introducir los datos de entrada. Expresa lectura
	Símbolo utilizado para representar un proceso. En su interior se expresan asignaciones, operaciones aritméticas, cambios de valor de celdas en memoria, etc.
	Símbolo utilizado para representar una decisión. En su interior se almacena una condición.
	Símbolo utilizado para representar una decisión múltiple.
	Representa la impresión de un resultado. Expresa escritura
	Expresa la dirección del flujo del diagrama
	Expresa conexión dentro de una misma página
	Expresa conexión dentro de diferentes páginas
	Expresa que para continuar con el flujo normal del diagrama debemos primero resolver el subproblema que anuncia en su interior

En el siguiente ejemplo se muestra el diagrama de flujo para un algoritmo que, al ingresar la edad, calcula si la persona es mayor o menor de edad.:



Otra forma de expresar un algoritmo, más cercana al lenguaje de programación, es en pseudocódigo ó pseudolenguaje, es decir en un "falso lenguaje". El pseudolenguaje tiene una estructura similar a la de un lenguaje de programación, pero incorpora arreglos del lenguaje hablado (es decir, no utiliza los códigos estrictos del lenguaje al cual remda). El ejemplo anterior puede escribirse de la siguiente forma en pseudolenguaje:

Algoritmo calcula_edad

Variables

Edad: entero

Inicio

Imprimir "Ingrese la edad"

Leer Edad

Si Edad $\geq$ 18 entonces

Imprimir "Eres mayor de edad"

Sino

Imprimir "Eres menor de edad"

Finsi

Fin

7. BASES DE DATOS

7.1. Concepto

Una Base (o Banco) de Datos es un conjunto de datos que pertenecen al mismo contexto, almacenados sistemáticamente para su posterior uso.

En este sentido una Biblioteca puede considerarse una base de datos compuesta, en su mayoría, por documentos y textos impresos en papel o indexados para su consulta. En la actualidad, y gracias al desarrollo tecnológico de campos como la Informática y la Electrónica, la mayoría de las bases de datos tienen formato electrónico, lo que ofrece un amplio rango de soluciones al problema de almacenar datos.

Si no existe una Base de Datos los datos se manejan a través de una colección de archivos, cuya administración se realiza a través de cada programa que los utiliza. Este es un enfoque de sistemas orientados hacia el proceso. Las aplicaciones se implementan con entera independencia unas de otras, y los datos no se suelen transferir entre ellas, sino que se duplican siempre que los correspondientes proyectos los necesitan. Este enfoque tiene, entre otras, las siguientes limitaciones:

- Hay una separación y aislamiento de los datos. Los usuarios de un programa pueden no estar enterados de los datos potencialmente útiles utilizados por otros programas.
- Hay una duplicación de datos, es decir, los mismos datos son mantenidos por diferentes programas. Esto conduce a espacio perdido y a valores y/o formatos potencialmente diferentes para el mismo dato.
- Las aplicaciones deben estar enteradas de la estructura del archivo. Si la estructura de archivo cambia, todos los programas que tienen acceso a esos datos también se deben modificar para reflejar los cambios. Las aplicaciones pueden tener que acceder datos que no necesitan mientras buscan alguna pieza de datos.
- Los formatos de archivos son incompatibles entre distintos programas, ya que la estructura de almacenamiento es definida en el código de la aplicación. Además, esa estructura es dependiente del lenguaje de programación de la aplicación.
- Los programas se escriben para satisfacer funciones particulares. Cualquier nuevo requisito necesita de un nuevo programa.
- No se pueden satisfacer reglas de Integridad, es decir, cada aplicación puede actualizar datos comunes sin que los otros lo hagan.

Las Bases de Datos se originaron para solucionar estos inconvenientes, a través de la creación de un software especializado que se encargue de realizar la administración de los datos de que hacen uso los software de aplicación. Los sistemas administradores de bases de datos (DBMS – Data Base Management System) permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las aplicaciones más usuales de DBMS son para la gestión de empresas e instituciones públicas. Pero también son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental.

7.2. DBMS

Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de una tarea específica en relación a la administración y organización de las bases de datos. El objetivo primordial de un DBMS es proporcionar un entorno que sea a la vez conveniente y eficiente para ser utilizado al extraer, almacenar y manipular información de la base de datos. Todas las peticiones de acceso a la base se manejan centralizadamente por medio del DBMS. El DBMS garantiza la integridad de los datos, es decir, impide que se introduzcan datos erróneos tanto por motivos físicos (defectos de hardware, actualización incompleta debido a causas externas) como de operación (introducción de datos incoherentes).

Un DBMS típico integra los siguientes componentes:

- Un lenguaje de definición de datos (DDL: Data Definition Language).
- Un lenguaje de manipulación de datos (DML: Data Manipulation Language)
- Un lenguaje de consulta (QL: Query Language).
- Una interfaz de usuario gráfica (GUI: Graphical User Interface).

Existen numerosos DBMS comerciales:

- Oracle (Oracle Corp.)
- Sybase (Sybase Inc.)
- SQL Server (Microsoft)
- Informix (actualmente pertenece a IBM)
- DB2 (IBM)
- Postgres
- Progress
- MySQL

BIBLIOGRAFÍA

- o STEWART, James. "Cálculo, Trascendentes Tempranas", 4 Ed., Tr. de Andrés Sestier. México, Ed. Thomson, 2002
- o Chiavenato, Idalberto. Introducción a la Teoría General de la Administración. 3ra. Edición. Edit. McGraw-Hill. 1992.
- o von Bertalanffy, Ludwig. Teoría General de Sistemas. Petrópolis, Vozes. 1976.
- o D. Patterson y J. Hennessy: "Computer Organization and Design: The Hardware/Software Interface". Segunda Edición. Morgan Kaufmann Publishers, Inc, 1998.
- o Joyanes, L. Aguilar fundamentos de programación, algoritmos, estructura de datos y objetos, Concepción Fernández Madrid, tercera edición, 2003.
- o Criado, M. Programación en lenguajes estructurados, ra-ma Madrid, 2006.
- o Joyanes, L.; Zahonero, I. Algoritmos y estructuras de datos una perspectiva en C, 2004.
- o Hernández, R.; Lázaro, J.C.; Dormido, R.; Ros, S. Estructuras de datos y algoritmos, Pearson Educación, 2001.
- o Aho. A.; Hopcroft, J.; Ullman. J. Estructuras de datos y algoritmos, Addison-Wesley, 1988.