

Tema: 1: Introducción a los sistemas informáticos y sistemas de numeración.

1. Introducción a los sistemas informáticos	1
2. Historia de los ordenadores	2
2.1. Primera Generación (1945-1958)	2
2.2. Segunda Generación (1958-1964)	2
2.3. Tercera Generación (1964-1971)	3
2.4. Cuarta Generación (1971-1983)	3
2.5. Quinta Generación (1983-1990)	3
2.6. Sexta generación (1990 hasta la fecha)	4
3. Tipos de sistemas informáticos	5
4. Componentes de los sistemas informáticos	6
5. Aplicaciones de los sistemas informáticos	7
6. Estructura física y funcional	8
6.1. Estructura física	8
6.2. Estructura funcional	9
7. Introducción a los Sistemas de Numeración	9
7.1. Sistemas de Numeración	10
7.2. Conversiones entre los sistemas de numeración: TFN	11
8. Sistemas de Codificación	14
8.1. Coma fija	15
8.1.1. Codificación en Signo y Magnitud (Módulo y signo)	15
8.2. Complemento a 1	15
8.3. Complemento a 2	15
8.4. Exceso a 2 elevado a (N-1)	16
8.5. COMA FLOTANTE	17
1.1.1. Simple precisión (4 bytes, 32 bits)	18
1.1.2. Doble precisión (8Bytes, 64 bits)	18
9. CÓDIGOS ALFANUMÉRICOS	19
9.1. Evolución histórica: ASCII, EBCDIC, UNICODE	19
9.2. Código UNICODE	23

1. Introducción a los sistemas informáticos.

Un sistema informático (SI) es un sistema que permite almacenar y procesar información; es el conjunto de partes interrelacionadas: hardware, software y personal informático. El hardware incluye computadoras o cualquier tipo de dispositivo electrónico, que consisten en procesadores, memoria, sistemas de almacenamiento externo, etc. El software incluye al sistema operativo, firmware y aplicaciones, siendo especialmente importante los sistemas de gestión de bases de datos. Por último, el soporte humano incluye al personal técnico que crean y mantienen el sistema (analistas, programadores, operarios, etc.) y a los usuarios que lo utilizan. Por ejemplo, una computadora, sus dispositivos periféricos y la persona que la maneja, pueden constituir un sistema informático.

Los Sistemas informáticos pueden clasificarse en base a numerosos criterios. Por supuesto las clasificaciones no son estancas y es común encontrar sistemas híbridos que no encajen en una única categoría.

Un sistema informático puede formar parte de un sistema de información (pero no es lo mismo); en este último, la información, uso y acceso a la misma, no necesariamente está informatizada. Por ejemplo, el sistema de archivo de libros de una biblioteca y su actividad en general es un sistema de información. Si dentro del sistema de información hay computadoras que ayudan en la tarea de organizar la biblioteca, entonces ese es un sistema informático.

2. Historia de los ordenadores.

2.1. Primera Generación (1945-1958).

Los primeros ordenadores se empezaron a desarrollar, durante la Segunda Guerra Mundial. En 1943, el ejército de Estados Unidos empezó a construir el ENIAC, Electronic Numerical Integrator And Compute. El propósito era utilizar la capacidad de cómputo para realizar cálculos de trayectorias balísticas. Este primer ordenador, estaba basado en el empleo de válvulas de vacío. La construcción duró hasta el año 1946, por lo que no pudo ser utilizado para su propósito inicial. En su lugar, fue empleado para realizar cálculos relacionados con la bomba de hidrógeno. Esto demostró la versatilidad de estos sistemas de propósito general. Uno de los principales inconvenientes del ENIAC era que tenía que ser programado manualmente. En esta generación había un gran desconocimiento de las capacidades de las computadoras. Esta generación abarcó la década de los cincuenta y se conoce como la primera generación. Estas máquinas tenían las siguientes características:

- Usaban válvulas de vacío para procesar información.
- Usaban tarjetas perforadas para introducir los datos y los programas.
- Usaban cilindros magnéticos para almacenar información e instrucciones internas.
- Eran sumamente grandes, utilizaban gran cantidad de electricidad, generaban gran cantidad de calor y eran sumamente lentas.
- Se comenzó a utilizar el sistema binario para representar los datos.
- En esta generación las máquinas son costosas (de un costo aproximado de 10,000 dólares).
- La computadora más exitosa de la primera generación fue la IBM 650.

Esta computadora usaba un esquema de memoria secundaria llamado tambor magnético, que es el antecesor de los discos actuales.



2.2. Segunda Generación (1958-1964).

En esta generación las computadoras se reducen de tamaño y son de menor costo. Aparecen muchas compañías y las computadoras eran bastante avanzadas para su época como la serie 5000 de Burroughs y la ATLAS de la Universidad de Manchester. Algunas computadoras se programaban con cintas perforadas y otras por medio de cableado en un tablero. Las características de esta generación eran:

- Usaban transistores para procesar información. Los transistores eran más rápidos, pequeños y confiables que los tubos de vacío.
- Usaban pequeños anillos magnéticos para almacenar información e instrucciones.
- Se mejoraron los programas de computadoras que fueron desarrollados durante la primera generación.
- Se desarrollaron nuevos lenguajes de programación como COBOL y FORTRAN, los cuales eran comercialmente accesibles.
- Se usaban en aplicaciones de sistemas de reservas de líneas aéreas, control del tráfico aéreo y simulaciones de propósito general.
- La marina de los Estados Unidos desarrolla el primer simulador de vuelo, "Whirlwind I".

- Surgieron las minicomputadoras y los terminales a distancia. □ Se comenzó a disminuir el tamaño de las computadoras.

2.3. Tercera Generación (1964-1971).

La tercera generación de computadoras emergió con el desarrollo de circuitos integrados (pastillas de silicio) en las que se colocan miles de componentes electrónicos en una integración en miniatura. Las computadoras nuevamente se hicieron más pequeñas, más rápidas, desprendían menos calor y eran energéticamente más eficientes. El ordenador IBM360 dominó las ventas de la tercera generación de ordenadores desde su presentación en 1965. El PDP-8 de la Digital Equipment Corporation fue el primer miniordenador. Las características de esta generación son:

- Se desarrollaron circuitos integrados para procesar información.
- Se desarrollaron los "chips" para almacenar y procesar la información. Un "chip" es una pieza de silicio que contiene los componentes electrónicos en miniatura llamados semiconductores.
- Los circuitos integrados recuerdan los datos, ya que almacenan la información como cargas eléctricas.
- Surge la multiprogramación.
- Las computadoras pueden llevar a cabo ambas tareas de procesamiento o análisis matemáticos.
- Emerge la industria del "software".
- Se desarrollan las minicomputadoras IBM 360 y DEC PDP-1.
- Otra vez las computadoras se tornan más pequeñas, más ligeras y más eficientes. □ Consumían menos electricidad, por lo tanto, generaban menos calor.

2.4. Cuarta Generación (1971-1983).

Aparecen los microprocesadores que es un gran adelanto de la microelectrónica, son circuitos integrados de alta densidad y con una velocidad impresionante. Las microcomputadoras basadas en estos circuitos son extremadamente pequeñas y baratas, por lo que su uso se extiende al mercado industrial. Aquí nacen los ordenadores personales que han adquirido proporciones enormes y que han influido en la sociedad en general sobre la llamada "revolución informática". Características de esta generación son:

- Se desarrolló el microprocesador.
- Se colocan más circuitos dentro de un "chip".
- "LSI - Large Scale Integration circuit".
- "VLSI - Very Large Scale Integration circuit".
- Cada "chip" puede hacer diferentes tareas.
- Un "chip" sencillo actualmente contiene la unidad de control y la unidad de aritmética/lógica. El tercer componente, la memoria primaria, es operado por otros "chips".
- Se reemplaza la memoria de anillos magnéticos por la memoria de "chips" de silicio.
- Se desarrollan las microcomputadoras, o sea, computadoras personales o PC. □ Se desarrollan las supercomputadoras.

2.5. Quinta Generación (1983-1990).

En vista de la acelerada marcha de la microelectrónica, la sociedad industrial se ha dado a la tarea de poner también a esa altura el desarrollo del software y los sistemas con que se manejan las computadoras. Surge la competencia internacional por el dominio del mercado de la computación, en la que se perfilan dos líderes que, sin embargo, no han podido alcanzar el nivel que se desea: la capacidad de comunicarse con la computadora en un lenguaje más cotidiano y no a través de códigos o lenguajes de control especializados.

Japón lanzó en 1983 el llamado "programa de la quinta generación de computadoras", con los objetivos explícitos de producir máquinas con innovaciones reales en los criterios mencionados. Y en los Estados Unidos ya está en actividad un programa en desarrollo que persigue objetivos semejantes, que pueden resumirse de la siguiente manera:

- Se desarrollan aún más las microcomputadoras y las supercomputadoras.
- Inteligencia artificial: es el campo de estudio que trata de aplicar los procesos del pensamiento humano usados en la solución de problemas a la computadora.
- Robótica: es el arte y ciencia de la creación y empleo de robots. Un robot es un sistema de computación híbrido independiente que realiza actividades físicas y de cálculo. Están siendo

diseñados con inteligencia artificial, para que puedan responder de manera más efectiva a situaciones no estructuradas.

- Sistemas expertos: aplicación de inteligencia artificial que usa una base de conocimiento de la experiencia humana para ayudar a la resolución de problemas.
- Redes de comunicaciones: los canales de comunicaciones que interconectan terminales y computadoras se conocen como redes de comunicaciones; todo el "hardware" que soporta las interconexiones y todo el "software" que administra la transmisión.

2.6. Sexta generación (1990 hasta la fecha).

Como supuestamente la sexta generación de computadoras está en marcha desde principios de los años noventa, debemos por lo menos, esbozar las características que deben tener las computadoras de esta generación. También se mencionan algunos de los avances tecnológicos de la última década del siglo XX y lo que se espera lograr a lo largo del siglo XXI. Las computadoras de esta generación cuentan con arquitecturas combinadas Paralelo / Vectorial, con cientos de microprocesadores vectoriales trabajando al mismo tiempo; se han creado computadoras capaces de realizar más de un millón de millones de operaciones aritméticas de punto flotante por segundo (teraflops); las redes de área mundial (Wide Area Network, WAN) seguirán creciendo desorbitadamente utilizando medios de comunicación a través de fibras ópticas y satélites, con anchos de banda impresionantes. Las tecnologías de esta generación ya han sido desarrolladas o están en ese proceso. Algunas de ellas son: inteligencia/artificial distribuida; teoría del caos, sistemas difusos, holografía, transistores ópticos, etcétera.

3. Tipos de sistemas informáticos.

Podemos encontrar distintas clasificaciones atendiendo a diferentes criterios para el concepto de Sistema Informático. Por ejemplo:

- Según el número de usuarios que pueden acceder a él de forma concurrente:
 - o Monousuario: es el que se caracteriza por el hecho de que es una sola persona la que está utilizando los recursos del sistema. Todo el hardware del sistema está a disposición de un solo usuario.
 - o Multiusuarios: dirigido a dos o más usuarios. En particular, dicese de los sistemas informáticos pensados para su aprovechamiento por múltiples usuarios, con preferencia trabajando en grupo.
- Según su Uso:
 - o De Uso General: son los que se utilizan para varios tipos de aplicaciones (es el caso de los ordenadores personales). o De Uso Específico: se caracterizan para ejecutar uno o unos pocos programas (es el caso de los robots industriales o de los videojuegos).
- Según sus Prestaciones y el tipo de computador:
 - o Workstation: en castellano, estaciones de trabajo, consisten en bases informáticas formadas por una o varias computadoras, que pertenecen o no a una misma red de trabajo. Son frecuentemente utilizadas en trabajos científicos o técnicos que requieren de procesamientos de información en serie o paralelos a grandes escalas.
 - o Terminales Ligeros: también conocidas como “Terminales Tontos” o “Thin Clients” éstos ordenadores solo poseen la capacidad de envío de datos y funcionan como complemento de otra central u ordenador. No es posible almacenar información ni procesarla sino generar nexos de datos.
 - o Microordenadores: ordenadores personales o de uso doméstico. Pueden tener o no como finalidad su utilización de manera profesional, pero están destinados al consumo general y no específico. Cuentan con la capacidad de almacenamiento y procesamiento de datos en escala media y también a la posibilidad de conectarse en red y a sistemas de redes.
 - o Macro ordenadores: computadoras centrales con las características de estar dotadas de una gran capacidad de almacenamiento y velocidad de procesamiento. Suelen ser utilizadas por grandes empresas que manejan enormes flujos de datos de manera simultánea. Generalmente cuentan con conexiones a ordenadores “satélites” o secundarios, donde los operadores van ejecutando diferentes órdenes y flujos de datos.
- Según la Arquitectura.
 - o Sistema Aislado: dispositivos que trabajan sólo con sus recursos locales, sin utilizar ningún recurso ya sea de una red local o de Internet.
 - o Arquitectura Cliente-Servidor: es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, que le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras. Los servidores debido a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema
 - o Sistemas distribuidos: en esta arquitectura la capacidad de proceso está repartida entre todos los equipos.
 - o Arquitectura de 3 Capas: en un sistema cliente servidor básico de dos capas generalmente los datos forman parte de la capa servidora. Separar en la capa servidora las aplicaciones de servidor de los datos permitiría independizar los servidores del medio utilizado para almacenar los datos, que podría ser una base de datos externa. Esto da lugar al sistema de tres capas donde una capa pertenece a las aplicaciones del cliente, otra a las aplicaciones servidoras y la tercera al medio de almacenamiento o generación de datos.
 - o Servidor de Aplicaciones: en informática, se denomina servidor de aplicaciones a un servidor en una red de computadores que ejecuta ciertas aplicaciones. Usualmente se trata de un dispositivo de software que proporciona servicios de aplicación a las

computadoras cliente. Un servidor de aplicaciones generalmente gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación. Los principales beneficios de la aplicación de la tecnología de servidores de aplicación son la centralización y la disminución de la complejidad en el desarrollo de aplicaciones.

- o Servidor de Transacciones o de Base de Datos: los servidores de bases de datos surgen con motivo de la necesidad de las empresas de manejar grandes y complejos volúmenes de datos, al tiempo que requieren compartir la información con un conjunto de clientes (que pueden ser tanto aplicaciones como usuarios) de una manera segura. Ante este enfoque, un sistema gestor de bases de datos (SGBD, a partir de ahora) deberá ofrecer soluciones de forma fiable, rentable y de alto rendimiento.

A estas tres características, le debemos añadir una más: debe proporcionar servicios de forma global y, en la medida de lo posible, independientemente de la plataforma.

- Según el Número de Tareas que pueden realizar de manera concurrente.
 - o Sistema monotarea: son aquellos que solo permiten realizar una sola tarea, estos sistemas son más primitivos porque solo permiten un solo proceso a la vez.
 - o Sistema multitarea: son sistemas que permiten realizar varias actividades o procesos a la vez sin ningún problema.
- Según el número de procesadores de los que disponen.
 - o Sistema monoprocesador: solo pueden trabajar con un procesador.
 - o Sistema multiproceso: pueden utilizar varios procesadores para distribuir el trabajo de cada uno. Generalmente estos trabajan o pueden ser de dos tipos:
 - ☐ Asimétrica: donde el sistema operativo selecciona un procesador maestro y los demás funcionan como esclavos.
 - ☐ Simétrica: en este tipo se envía información o se trabaja con el procesador con menos carga y así se distribuye mejor el trabajo, los procesos son enviados indistintamente a cualquiera de los procesadores disponibles.

4. Componentes de los sistemas informáticos.

Se consideran componentes del sistema informático:

- Personas en los dos extremos de la cadena de procesamiento físico de la información: en la realización de los hechos que generan los datos primarios y su captación y adquisición y en la consulta y utilización de ellos. Por supuesto también intervienen en ciertas acciones de operación del sistema. Incluye tanto a los técnicos, como a los usuarios.
- Hardware: equipos de procesamiento de información, fundamentalmente computadoras.
- Equipos de apoyo a las transmisiones (HUB, gateways), equipos de apoyo y de seguridad (back-ups, acondicionadores de aire, deshumificadores, entre otros).
- Software: programas de computadoras, sistemas operativos, programas de servicio de comunicaciones, y sobre todo, programas de aplicación).
- Información técnica de apoyo al sistema: manuales técnicos sobre el trabajo de las computadoras y los equipos de apoyo, manuales técnicos sobre los sistemas operativos y programas generales.
- Manuales de usuario: para orientar a los usuarios-operadores sobre su trabajo con el sistema de información. Incluyen la definición de los procedimientos manuales que deben realizar los usuarios-operadores, la descripción de los formularios para captar la información primaria, la descripción de los reportes de salida y la descripción de las acciones interactivas con el sistema informático: captación de la información, operación del mismo, acciones ante errores y situaciones anormales, seguridad y protección de los recursos informativos y consulta de información de resultados.
- Informaciones de todo tipo, soportadas sobre formularios de papel, CDs, DVDs, reportes de papel de impresora, bases de datos en línea almacenadas en discos duros.

5. Aplicaciones de los sistemas informáticos.

- Sistemas de procesamiento básico de la información: aquellos en que las computadoras se limitan a realizar las operaciones de procesamiento físico de la información. Las personas que integran el sistema, asumen todas las labores de generación de la información primaria y de análisis de información de resultados.
 - o Sistemas de procesamiento de transacciones (TPS): proceso físico de los datos relacionados con ciertas transacciones rutinarias y aisladas en el trabajo habitual de las entidades socioeconómicas, tales como el control de inventarios, control de activos fijos o la nómina de sueldos o salarios, explotan poco las posibilidades de las máquinas y el software actual.
 - o Sistemas de automatización de oficinas (OAS): incluye el empleo de procesadores de texto, hojas electrónicas de datos, preparadores de exposiciones, calendarización, comunicación mediante correos electrónicos, videoconferencias, implican la búsqueda y captación de operaciones y en muchos casos, la preparación de decisiones para ejecutivos y directivos. Pueden solucionar tareas típicas de las oficinas, como la programación y control de actividades mediante agendas electrónicas individuales y colectivas, registro y control de acuerdos y directrices, escritura y conformación de textos en informes, folletos, creación, actualización y consulta de bases de datos relacionadas con clientes y vendedores.
 - o Sistemas de información para la dirección (MIS): estos sistemas han abarcado los TPS, integrando las mismas mediante sistemas de bases de datos, y almacenes de datos, de forma tal que el sistema puede reflejar la realidad compleja de una entidad socioeconómica, con todos sus subsistemas y relaciones informativas. Se orientan, sobre todo, a proporcionar información para la toma de decisiones y el control, por lo que puede asegurarse que el rol de la computadora en estos sistemas es relativamente pasivo.
- Sistemas de apoyo a la toma de decisiones: se apoyan en los MIS, los que crean y actualizan las bases de datos, que los primeros utilizan. Los DSS se destinan a la toma de decisiones, están hechos para apoyar el trabajo individual o para las decisiones en grupo, apoyan mucho en la llamada investigación de operaciones o los métodos cuantitativo de la toma de decisiones, técnicas matemáticas para apoyar el trabajo del ser humano en las llamadas decisiones bien estructuradas, débilmente estructuradas y no estructuradas, las cuales por su complejidad pueden tener errores al ser analizadas por el ser humano con métodos tradicionales (intuición, experiencia). Ejemplo de estas decisiones son el empleo de técnicas de ruta crítica para dirigir proyectos de construcciones.
- Sistemas basados en la inteligencia artificial: la inteligencia artificial, es una rama de la ciencia de la computación que busca emular las capacidades intelectuales del ser humano, mediante el empleo de software especializado y las computadoras. Abarca muchos campos, entre los que se encuentra la robótica, la solución general de problemas, identificación y reconocimiento de patrones visuales, auditivos y digitales, la simulación del movimiento, el análisis y la síntesis del lenguaje natural y la potenciación del conocimiento humano. Representan un paso adelante en relación con los anteriores MIS y DDS, pueden asumir actividades más “humanas”, más activas en los procesos de dirección, pues tiene incorporados muchos elementos que los hacen actuar similarmente como lo haría un humano.
- Sistemas de expertos o basados en las reglas de conocimientos: se basan en disponer del conocimiento de uno o más expertos humanos, por lo general en forma de reglas de producción, expresadas en forma de IF (condición), THEN(acción), ELSE (acción alternativa), SI (condición), entonces ejecutar a (acción) y en caso contrario ejecutar la (acción alternativa), para la solución de un problema concreto determinado.
- Sistemas de razonamiento basado en casos (RBC): se basan en la analogía como forma de representación del conocimiento, se fundamentan en la aparente forma humana de razonar, usan las experiencias pasadas para afrontar problemas nuevos, consecuentemente aplican técnicas de intuición y permiten el aprendizaje. Su base teórica se apoya en el campo del razonamiento aproximado. Su filosofía implica aprender de los errores y los éxitos anteriores. Pueden utilizarse con éxito en problemas de interpretación, justificación, clasificación, proyección y previsión. Ejemplo: JULIA: Diseño de grupos de comidas.
- Sistemas de redes neuronales artificiales (RNA): busca simular la forma en que supuestamente trabajan las neuronas naturales en el cerebro humano. Cada sistema está compuesto de muchas

neuronas artificiales, las cuales, al asociarse entre sí, mediante determinadas arquitecturas pueden realizar trabajos complejos con asombrosa calidad. Su mayor virtud es que logran trabajar adecuadamente, aunque la información de entrada que reciban tenga cierto grado de errores o sea hasta cierto punto incompleto. Pero la asociación de esas neuronas artificiales hace que trabajos simples se conviertan en actividades muy complejas que se realizan con éxito. Ejemplo: diseño de un RNA para realizar evaluaciones del sistema de control interno en determinadas entidades.

- Sistemas basados en algoritmos genéticos (GA): conjunto de métodos que se apoyan, como su nombre lo indica, en la emulación de la forma de desarrollo genético de los animales y las plantas, estos son entre sí mismos una técnica de aprendizaje automático, aspiran a permitir que las computadoras “aprendan” por sí mismas, para algunos especialistas son solo una familia de métodos de búsqueda adaptativa de soluciones.
- Sistemas basados en técnicas web: el auge en el mundo de uno de los servicios de la Internet, el World Wide Web (W.W.W) ha hecho que surgiera una nueva modalidad de sistemas a partir de la tecnología WEB, y en el concepto de hipertexto e hipermedia.
 - o Intranets: surge de la utilización de la WEB en la gestión interna de la entidad. Una intranet es una red particular, basada en redes de comunicación de área local o en redes de área amplia, que utiliza tecnología estándar y servicios o productos que se pueden encontrar o han sido desarrollados para Internet. Una Intranet puede tener o no conexión con Internet y en caso de tenerla puede filtrar los accesos no deseados con los llamados cortafuegos.
 - o Los sistemas basados en la WEB, pueden ser también de uso externo, o sea, para comunicar información al entorno de la entidad (clientes, suministradores, niveles superiores, agencias gubernamentales, público en general y otras entidades políticas o administrativas de control). En estos casos la información que aparecerá en el sitio WEB estará acorde con la misión y los objetivos de la entidad.
- Sistemas de gestión del conocimiento: es conocido que el conocimiento está asociado a la información, pero es mucho más, el conocimiento es, desde la perspectiva de la ciencia de la dirección, la capacidad de solucionar problemas, el saber qué hacer, cómo hacerlo, dónde hacerlo, y para qué hacerlo. Este conocimiento da una ventaja competitiva a las entidades que lo poseen, por lo tanto, debe ser protegido, pero también debe ser conservado y difundido entre los miembros de una entidad, para que la pericia de unos, pase a ser patrimonio de otros. En esto consiste la gestión del conocimiento.
- Los sistemas que se encargan de esa gestión, o sea, detección, adquisición, conservación y difusión del conocimiento son los sistemas de gestión del conocimiento. Dos tipos de sistemas que utilizan las entidades socioeconómicas son:
 - o Sistemas de información de marketing (SIM): se destinan a gestionar toda la información con el mercado, la gestión de clientes potenciales y reales, los estudios de mercado, el análisis del impacto de las políticas de promoción, el monitoreo y análisis de los competidores, el análisis de los precios, el estudio de las tendencias de consumo, los estudios macroeconómicos de tendencias demográficas.
 - o Sistemas de gestión de relaciones (e-CRM): son sistemas muy asociados a los SIM y a la gestión comercial, pues se utilizan para propiciar una adecuada relación con los clientes de la entidad. Se utilizan prácticamente en todo el ciclo de relaciones con el cliente. Pueden emplearse para definir: provisiones de ventas, registros de visitas de gestión al cliente, contactos realizados en ferias y congresos, volúmenes de compras anteriores, intenciones de compra anterior o satisfechas, comportamiento de pago, bancos con los que trabaja, oportunidades de negocio, acciones directas de marketing que ha recibido.

6. Estructura física y funcional.

6.1. Estructura física.

Cualquier sistema informático, excepto aquellos formados por un único ordenador aislado, está compuesto por una serie de dispositivos, cada uno de ellos conectado al menos con otro de forma que puedan comunicarse entre sí. Esta conexión de múltiples dispositivos puede realizarse de diversas maneras, lo que da lugar a distintas estructuras físicas para cada uno de estos sistemas.

Las estructuras físicas más comúnmente usadas son las siguientes:

- Una Topología de bus usa un solo cable (Backbone) que debe terminarse en ambos extremos. Todos los Hosts se conectan directamente a este Backbone. La topología en bus es una topología de red multipunto, en la cual los dispositivos se conectan a un mismo cable, uno tras otro. En la topología en bus, todos los dispositivos comparten el mismo medio, por esta razón, los mensajes que se transmiten a través de este son atendidos por todos los demás dispositivos que lo comparten.
- La Topología de anillo conecta un host con el siguiente y al último host con el primero. Esto crea un anillo físico de cable. el mensaje viaja en una sola dirección y es leído por cada una de las computadoras individualmente y retransmitido al anillo en caso de no ser el destinatario final de los mensajes.
- La Topología en estrella conecta todos los cables con un punto central de concentración.
- Una Topología en estrella extendida: conecta estrellas individuales entre sí mediante la conexión de Hubs o Switches. Esta topología puede extender el alcance y la cobertura de la red.
- Una Topología jerárquica es similar a una estrella extendida. Pero en lugar de conectar los Hubs o Switches entre sí, el sistema se conecta con un computador que controla el tráfico de la topología.
- La Topología de malla se implementa para proporcionar la mayor protección posible para evitar una interrupción del servicio. En esta topología, cada host tiene sus propias conexiones con los demás hosts. Aunque Internet cuenta con múltiples rutas hacia cualquier ubicación, no adopta la topología de malla completa.
Esta topología, a diferencia de otras (como la topología en árbol y la topología en estrella), no requiere de un servidor o nodo central, con lo que se reduce el mantenimiento.
- La Topología de árbol tiene varias terminales conectadas de forma que la red se ramifica desde un Servidor base. La red en árbol es una topología de red en la que los nodos están colocados en forma de árbol. Desde una visión topológica, es parecida a una serie de redes en estrella interconectadas salvo en que no tiene un nodo central.

6.2. Estructura funcional.

Por estructura funcional entendemos la manera en la que los distintos elementos del sistema colaboran entre sí para realizar las tareas asignadas por el usuario.

- Cliente-Servidor: los procesos pueden ser tanto servidores como clientes. El sistema informático mantiene la comunicación entre procesos. Ofrece una gran flexibilidad en cuanto a los servicios del sistema final.
- Sistemas centralizados: se trata de un gran ordenador (mainframe) que realiza todo el procesamiento y los usuarios se conectan a él mediante terminales tontos (sin capacidad de procesamiento) que no disponen de memoria, ni de procesador.
- Sistemas en red: estos sistemas comunican varios ordenadores entre sí por algún medio de comunicación con el fin de compartir los recursos y la información del sistema. Cada ordenador posee su propio sistema operativo y su sistema de ficheros local.
- Sistemas distribuidos: es una variante del sistema en red pero integrando los recursos en una máquina virtual de manera que el usuario accede de forma transparente sin saber dónde se encuentran los recursos; permite distribuir los trabajos, tareas o procesos entre un conjunto de procesadores. Las principales ventajas de los sistemas distribuidos son: compartición de recursos, aceleración de los cálculos, fiabilidad, comunicación y sistemas no heterogéneos, logrando mediante los servicios de los de red, integrar recursos (impresoras, unidades de almacenamiento, memoria, procesos, CPUs) en una sola máquina virtual que el usuario accede de forma transparente.

7. Introducción a los Sistemas de Numeración.

El problema que nos ocupa es la representación de la información para su almacenamiento y procesamiento por un sistema informático.

Sabemos que, el ser humano en su vida cotidiana trabaja con el sistema decimal desde el punto de vista numérico y desde el punto de vista alfabético con un determinado idioma; y que además, utiliza otros muchos códigos y símbolos: señales de tráfico, etc.

El ordenador, por cuestiones de índole técnica, lo tiene que hacer todo utilizando el sistema binario. El sistema binario utiliza dos únicos símbolos para su representación, el 0 y el 1, lo cual se adapta

perfectamente a la particularidad de los elementos electrónicos. Los circuitos electrónicos de un ordenador están constituidos, en su mayoría, por elementos de tipo biestable - condensadores, diodos, transistores, etc - que solo permiten dos estados estables, hay carga eléctrica o no la hay, deja pasar la corriente o no la deja pasar, etc. En los circuitos electrónicos se suele asociar la presencia de tensión en un punto de un circuito por medio de un 1 y la ausencia de tensión por medio de un 0. Cada uno de estos unos o ceros recibe el nombre de bit.

Así pues, es necesario que todos los métodos de codificación internos del ordenador tengan su origen en el sistema binario, y utilizar distintas codificaciones para poder representar todo tipo de informaciones que maneje una computadora:

- Información numérica con la que ordenador sea capaz de operar
- Información alfanumérica (todo tipo de símbolos, letras mayúsculas y minúsculas, dígitos, etc)
- Códigos de control en las transmisiones de datos.
- Códigos de protección y confidencialidad de los datos.
- Códigos para almacenar imágenes, secuencias de video, etc
- Códigos de comprensión para reducir el tamaño de almacenamiento de los datos.
- Etc ...

Antes de pasar a estudiar algunos de estos códigos de representación de datos, vamos a familiarizarnos con los procedimientos básicos para tratar con distintos sistemas de numeración, haciendo especial hincapié en los más utilizados en informática: el decimal, el binario, el octal y el hexadecimal.

7.1. Sistemas de Numeración.

Definimos sistema de numeración, como el conjunto de símbolos y reglas utilizados para la representación de cantidades numéricas.

Los sistemas de numeración utilizados en la actualidad se caracterizan por dos aspectos:

En primer lugar, se distinguen por su base. Lo cual determina el número de símbolos que puede utilizar el correspondiente sistema de numeración y que viene dado por la siguiente regla:

Desde el símbolo 0 hasta el símbolo (Base -1)

DECIMAL (B=10)	10 símbolos diferentes para representar las cantidades. Estos símbolos van desde el 0 hasta (10-1): 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
BINARIO (B=2)	2 símbolos diferentes, estos son los dígitos 1 y 0. A cada uno de estos símbolos se le denomina bit (binary digit).
OCTAL (B=8)	Utiliza 8 símbolos para la representación de las cantidades. Estos símbolos son: 0 1 2 3 4 5 6 7
HEXADECIMAL (B=16)	Utiliza 16 símbolos para la representación de cantidades. Estos símbolos son: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. A los símbolos A, B, C, D, E, F se le asignan los siguientes valores absolutos: 10, 11, 12, 13, 14 y 15.

En segundo lugar, son sistemas de numeración posicionales. Esto quiere decir que cada uno de sus símbolos puede adoptar distintos valores según la posición que ocupe dentro de la cantidad representada. Por ejemplo, el símbolo 2 del sistema decimal puede valer 2, 20, 200 según ocupe la posición de las unidades, de las decenas, de las centenas, etc. La base es el coeficiente que determina el valor de cada posición. La posición será el exponente de dicho coeficiente y se establece con relación al símbolo "coma decimal". La "coma decimal", si no aparece, se supone colocada implícitamente a la derecha. Las posiciones a la izquierda de la coma decimal se numeran a partir de 0 y las de la derecha a partir de -1.

En resumen, la BASE determina los dos aspectos fundamentales de los sistemas de numeración actuales:

- su número de símbolos
- y el valor de las distintas posiciones.

7.2. Conversiones entre los sistemas de numeración: TFN.

Los procedimientos de conversión nos permiten calcular o conocer a qué valor corresponde una cantidad representada en un sistema de numeración B en cualquier otro sistema de numeración B', y viceversa.

Vamos a repasar todas las situaciones generales posibles y dentro de ellas algunas situaciones especiales:

De cualquier base B --> a base Decimal De base

Decimal --> a cualquier base B

De cualquier base B --> a cualquier base B'

Para los sistemas de numeración más comunes en informática (decimal, binario, octal y hexadecimal) veremos una forma rápida o especial de conversión entre ellos.

Pasar de cualquier base B --> a base Decimal

El Teorema Fundamental de la Numeración (TFN), relaciona una cantidad expresada en cualquier sistema de numeración (base B) con la misma cantidad expresada en el sistema decimal (base 10).

A. Procedimiento general:

Suponiendo una cantidad X expresada en un sistema cuya base es B y que cada uno de los dígitos que contiene dicha cantidad los representamos por X_i , donde el subíndice (i) indica la posición del dígito con respecto a la coma decimal. Hacia la izquierda de la coma se numera desde 0 en adelante de 1 en 1 (n n-1 ... 3 2 1 0), y hacia la derecha se numera desde -1 y con incremento de -1 (-1 -2 -3 ... -(n-1) -n). El TFN dice que el valor decimal de esta cantidad X, vendrá dado por la siguiente fórmula:

$$\dots + X_4 \cdot B^4 + X_3 \cdot B^3 + X_2 \cdot B^2 + X_1 \cdot B^1 + X_0 \cdot B^0 + X_{-1} \cdot B^{-1} + X_{-2} \cdot B^{-2} + \dots$$

B. Caso especial: Pasar de Binario a Decimal.

La conversión del Sistema Binario a Decimal es rápida si conocemos el valor de cada una las posiciones, las cuales se duplican en cada posición a la izquierda de la coma decimal y se disminuyen a la mitad en cada posición a la derecha de la coma decimal. En este caso, sólo tendremos que sumar el valor de las posiciones en las que tengamos el dígito 1 obviando las demás. Por ejemplo: ¿Qué número decimal representa el número binario 1001101,011?

Posiciones	7	6	5	4	3	2	1	0	-1	-2	-3	-4
	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
Valor	128	64	32	16	8	4	2	1	0,5	0,25	0,125	0,0625
Ejemplo		1	0	0	1	1	0	1,	0	1	1	

Por tanto, basta con sumar: $64 + 8 + 4 + 1 + 0,25 + 0,125 = 77,375$

Pasar de Decimal --> a cualquier base B

El TFN aplicado de forma inversa nos permite relacionar una cantidad expresada en el sistema de numeración Decimal con la misma cantidad expresada en cualquier otro sistema de numeración (base B).

A. Procedimiento general:

El procedimiento es diferente para la parte entera y para la parte fraccionaria:

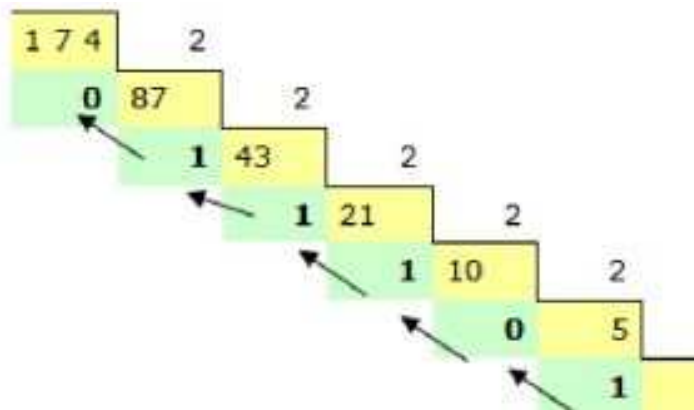
Parte entera: Se realiza mediante divisiones sucesivas por la base B a la que la queremos convertir. Los restos de dividir la cifra decimal entre la base B, serán los coeficientes del número, hasta que el cociente sea menor que el divisor (la base). El resultado vendrá dado por el último cociente más todos los restos obtenidos escritos en orden inverso, esto es, el primer dígito de la izquierda será el último cociente obtenido y el primer dígito de la derecha será el primer resto obtenido.

Parte fraccionaria: Si la cantidad expresada en base 10 tiene parte fraccionaria, para pasarla a un sistema de base “B” se realiza mediante multiplicaciones sucesivas de la parte fraccionaria por la base B. Básicamente se siguen los siguientes pasos:

1. A la cantidad correspondiente le restamos su parte entera.
2. El resultado (parte fraccionaria decimal) se multiplica por “B”.
3. La parte entera del resultado anterior, es la primera cifra de la parte fraccionaria en base “B”.
4. Si ya no hay parte fraccionaria, se acabó. Si aún queda parte fraccionaria, se vuelve al punto 1 con el valor obtenido en el paso anterior.

EJEMPLO:

La cantidad 174 que está en base 10 la queremos convertir a base 2.



B. Caso especial: Pasar de Decimal a Binario.

Tal como hemos visto anteriormente, la conversión del Sistema Decimal a Binario también es rápido si conocemos el valor de cada una de las posiciones.

En este caso, pondremos el primer 1 en la posición cuyo valor se aproxime más (sin pasarlo) al valor decimal que quiero convertir. Continuaré poniendo 1 en las posiciones a su derecha que vayan sumando sus valores hasta completar la cantidad a convertir. El resto de las posiciones hay que escribir el 0.

EJEMPLO: ¿Qué cantidad binaria representa el número decimal 47,75?

7	6	5	4	3	2	1	0	-1	-2	-3	-4
128	64	32	16	8	4	2	1	0,5	0,25	0,125	0,0625
		1	0	1	1	1	1	1	1		

Empezamos poniendo un 1 en la posición cuyo valor es 32, ya que la siguiente (64) se pasa. Por tanto, ahora nos queda $47,75 - 32 = 15,75$. Ponemos un 1 en el 8 que es más cercano sin pasarnos. Y así sucesivamente. Resultado: 101111,11

B. Casos especiales: Conversiones entre los sistemas Binario/Octal/Hexadecimal

Binario => Octal

Octal => Binario

Hexadecimal => Binario

Binario => Hexadecimal

Octal => Hexadecimal

Hexadecimal => Octal

Cuando los sistemas de numeración B y B' son potencias de dos, tal como sucede con los sistemas Binario (2^1), Octal (2^3) y Hexadecimal (2^4), la conversión entre ellos tendrá pasos directos mediante su correspondencia de dígitos binarios:

Cada dígito octal ($8=2^3$) se representará con su correspondiente combinación binaria de 3 bits (lo que marca el exponente).

Cada dígito hexadecimal ($16=2^4$) se representará con su correspondiente combinación binaria de 4 bits.

Los bits se agruparán siempre partiendo de la coma decimal hacia la izquierda (parte entera) o hacia la derecha (parte fraccionaria).

Dígito octal	Dígitos binarios	Dígito Hexadecimal	Dígitos Binarios
0	000	0	0000
1	001	1	0001
2	010	2	0010
3	011	3	0011
4	100	4	0100
5	101	5	0101
6	110	6	0110
7	111	7	0111
		8	1000
		9	1001
		A	1010
		B	1011
		C	1100
		D	1101
		E	1110
		F	1111

EJEMPLOS:

1) Pasar el número hexadecimal 2BC a binario.

2 B C.

0010 1011 1100 Luego $2BC_{(16)} = 1010111100_2$

2) Convertir el número binario 100101100 a hexadecimal:

0001 0010 1100

1 2 C Por tanto: $100101100_2 = 12C_{(16)}$

Convertir el número octal 75643,57 a binario.

7 5 6 4 3 5 7

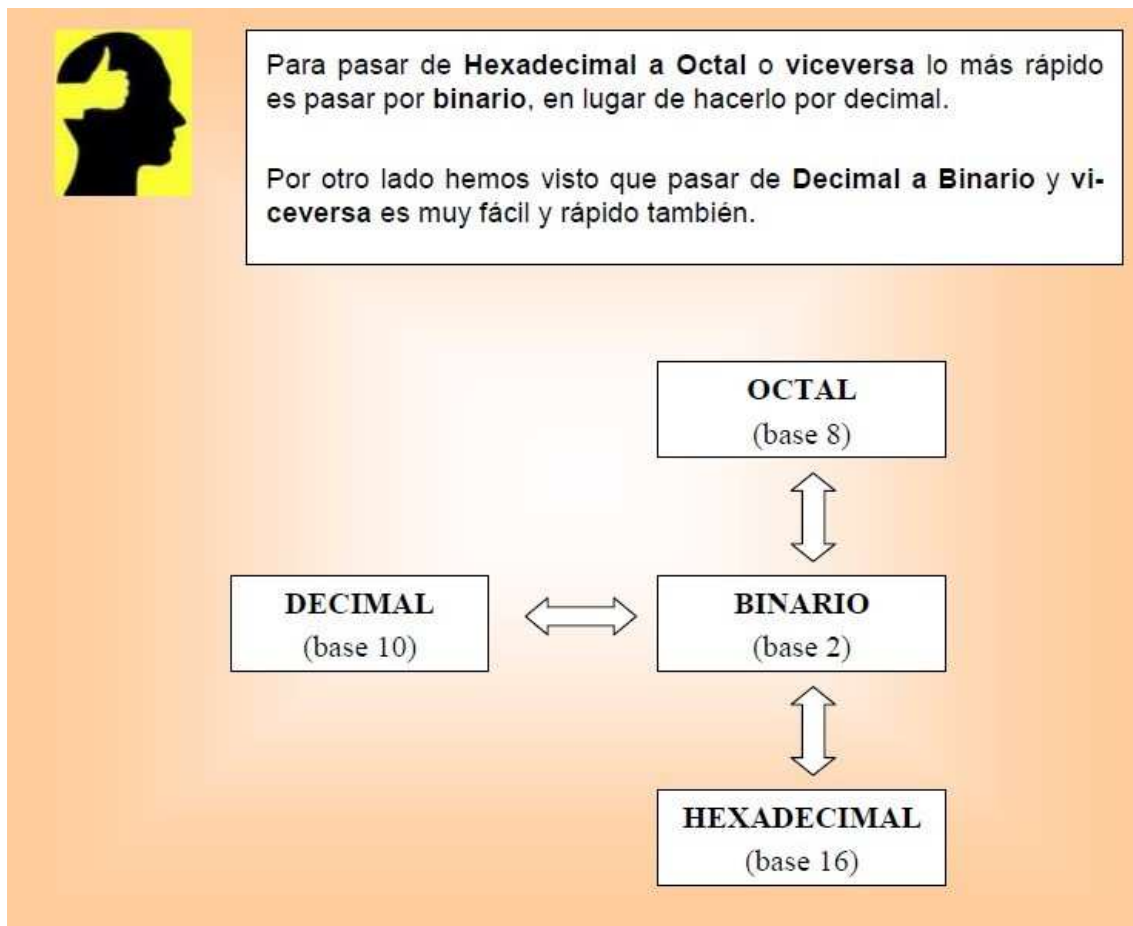
111 101 110 100 011, 101 111 Luego $75643,57_8 = 111101110100011,101111_2$

4) Convertir el número binario 1010111100 a octal.

001 010 111 100

1 2 7 4

Por tanto: $1010111100_2 = 1274_8$



8. Sistemas de Codificación.

Ya sabemos manejarnos con los sistemas de numeración más utilizados en informática, ahora vamos a ver como se agrupan una secuencia de bits para representar un determinado tipo de datos (numéricos, alfanuméricos, de control de transmisión, etc).

CÓDIGOS NUMÉRICOS

Hasta ahora hemos hablado de números enteros y fraccionarios en distintos sistemas de numeración, pero no lo hemos hecho de los números negativos. Al considerar los negativos unificamos las operaciones de adición y sustracción, dado que la resta no es más que la suma cambiando el signo del segundo sumando (minuyendo): $A - B = A + (-B)$.

Para indicar que un número es negativo añadimos el símbolo '-'. Pero el ordenador sólo trabaja con 0 y 1, por tanto tendremos que pensar en como indicarle al ordenador que un número es negativo. Así pues, todos los sistemas de codificación de números definen como deben representarse los números positivos y negativos, los números enteros y los fraccionarios.

Dependiendo del tipo de número que se quiera representar, las operaciones aritméticas que se efectúen y el rango a representar nos podemos encontrar con los siguientes tipos o códigos de representación numérica:

- Codificación en Signo y Magnitud (o Módulo y Signo).
- Codificación en Complemento a 1.
- Codificación en Complemento a 2.
- Exceso en 2^{n-1} .
- Codificación en Coma Flotante.

Es primordial hacer constar que, en cualquiera de los métodos apuntados, disponemos de un número limitado y fijo de bits para cada dato que se vaya a representar ya que la capacidad del ordenador es finita. Este número de bits lo representaremos por N en los siguientes apartados.

Se denomina Rango de Representación al conjunto de números que es capaz de representar una determinada codificación. El rango de representación de cualquier método de codificación binaria dependerá del número de bits reservados para su representación interna.

8.1. Coma fija.

8.1.1. Codificación en Signo y Magnitud (Módulo y signo)

En este sistema de codificación el bit situado más a la izquierda representa el Signo, y su valor será:

0 para los números positivos (+)

1 para los negativos (-)

El resto de los bits (N-1) representan el Módulo del número. Este vendrá dado por su correspondiente valor binario.

EJEMPLO: Supongamos que el número de bits reservados para la codificación de números enteros es de 8 bits (N=8) y queremos codificar los números +10 y -10

+10 sería 0 000 1010

-10 sería 1 000 1010

En el ejemplo anterior (N=8), su rango de representación es: de - 127 a + 127

$$(-2^{N-1} + 1) \text{ a } (+2^{N-1} - 1)$$

¿Ventaja?: Rango simétrico

Inconveniente muy importante: El valor cero tiene dos representaciones distintas.

Si N=8, +0 => 0000 0000

-0 => 1000 0000

8.2. Complemento a 1.

Números positivos: su codificación es exactamente igual que la vista en Módulo y Signo. La diferencia está en la representación de los números negativos.

Números negativos: su codificación se obtiene complementando todos los dígitos de su correspondiente codificación positiva; esto es, cambiando los ceros por unos y viceversa, incluido el bit de signo.

EJEMPLO: Codificar en complemento a 1 el número negativo -12, siendo N=8. +12

sería 0 0 0 0 1 1 0 0 -12 es 1 1 1 1 0 0 1 1

El rango de representación de este método es el mismo que para Módulo y Signo, con sus mismas ¿ventajas? e inconvenientes.

- Para 8 bits: $-127 \leq x \leq 127$

- Para 16 bits: $-32767 \leq x \leq 32767$

- Para 32 bits: $-2147483647 \leq x \leq 2147483647$

Dos representaciones distintas para el 0: Si N=8,

+0 □□ 0000 0000

-0 □□ 1111 1111

8.3. Complemento a 2.

Números positivos: su codificación es exactamente igual que la vista en Módulo y Signo y Complemento a 1. La diferencia está en la representación de los números negativos.

Números negativos: su codificación se obtiene en tres pasos:

1º Obtener su correspondiente número positivo.

2º Complementar ceros por unos y viceversa (complemento a 1)

3º Al complemento a 1 le sumamos 1, despreciando el último acarreo si existe.

EJEMPLO: Codificar en complemento a 2 el número negativo -12, siendo N=8. +12 sería 0

0 0 0 1 1 0 0

-12 complemento a 1 es 1 1 1 1 0 0 1 1

+ 1

-12 complemento a 2 es 1 1 1 1 0 1 0 0

Ventaja Importante el cero tiene una única representación. Veámosla para 8 bits:

Número 0 □□ 0 0 0 0 0 0 0 0

Número -0 □□ Primer paso: 1 1 1 1 1 1 1 1

Segundo paso: 1 1 1 1 1 1 1

Tercer paso + 1

1 0 0 0 0 0 0 0 Se desprecia el 1 acarreo

Por esta razón, el rango de representación de este método es asimétrico, y como la representación del cero la tomamos de los positivos, estos tendrán un número menos:

- 2^{N-1} a $2^{N-1} - 1$
- Para 8 bits: -128 a +127
- Para 16 bits: -32.768 a 32.767
- Para 32 bits: -2.147.483.648 a 2.147.483.647

8.4. Exceso a 2 elevado a (N-1).

Esta forma de representar los enteros que, al igual que los métodos anteriores, incluye a los negativos, se utiliza en la codificación de los exponentes (coma flotante, exponentes sesgados).

Los números se incrementan en un valor 2 elevado a N-1 al representarse en binario.

Así, el número 2^{N-1} representa al cero, los menores de 2^{N-1} a los negativos, y los mayores de 2^{N-1} a los positivos.

Este método no utiliza, implícitamente, ningún bit para el signo, con lo cual todos los bits representan un módulo o valor que corresponde al número representado más el exceso (el sesgo), que para N bits viene dado por 2 elevado a N-1.

Tanto si el número a representar es positivo como si es negativo los pasos a seguir son:

1° Sabiendo el valor de N, calcular el exceso (sesgo).

2° Sumar el exceso al número decimal que queremos codificar.

3° Convertir el número decimal a binario y, en caso necesario, completar con ceros a la izquierda hasta alcanzar las N posiciones establecidas.

Por ejemplo, para 8 bits el exceso es de 128, con lo que el número 10 vendrá representado por $10 + 128 = 138$; para el caso de -10 tendremos $-10 + 128 = 118$.

Veamos cuáles son sus representaciones:

Número 10 1 0 0 0 1 0 1 0 (138 en base 2) Número -10
0 1 1 1 0 1 1 0 (118 en base 2)

En este caso, el 0 tiene una única representación, que para 8 bits corresponde a:

Número ± 0 ($0 + 128$) = 1 0 0 0 0 0 0 0

El rango de representación es asimétrico y viene dado por:

- 2^{N-1} a $2^{N-1} - 1$
- Para 8 bits: $-128 \leq x \leq 127$
- Para 16 bits: $-32768 \leq x \leq 32767$
- Para 32 bits: $-2147483648 \leq x \leq 2147483647$

Para 8 bits, el valor más positivo representable es 127, que al sumarle 128, quedaría 255.

NOTA: Resulta interesante observar que todo número representado en exceso es igual a su correspondiente representación en complemento a 2 con el primer dígito de la izquierda cambiado.

Ejemplos:

a) Codificar en Exceso a 2^{N-1} el número negativo -12, siendo N=8.

Exceso = $2^{8-1} = 2^7 = 128$

$-12 + 128 = + 116$

116 en binario es 0 1 1 1 0 1 0 0

-12 codificado en Exceso, siendo N=8 es 0 1 1 1 0 1 0 0

b) Codificar en Exceso a 2^{N-1} el número positivo +12, siendo N=8.

Exceso = $2^{8-1} = 2^7 = 128$

$+12 + 128 = + 140$

140 en binario es 1 0 0 0 1 1 0 0

+12 codificado en Exceso, siendo N=8 es 1 0 0 0 1 1 0 0

8.5. COMA FLOTANTE

La coma flotante surge de la necesidad de representar números reales con un rango de representación mayor al que nos ofrece la representación en coma fija y posibilitar a la computadora el tratamiento de números muy grandes y muy pequeños.

En un sistema informático, será necesario representar números grandes como la constante de Avogadro ($6,022 \cdot 10^{23}$), serían necesarias 80 cifras en código binario, como números pequeños como la carga del electrón (1.60210^{-19}), que en binario natural serían necesarios más de 63 cifras para tener un valor significativo.

Para ello se utiliza la notación matemática exponencial, según la cual:

$$N^{\circ} = \text{MANTISA} * \text{BASE}^{\text{EXPONENTE}}$$

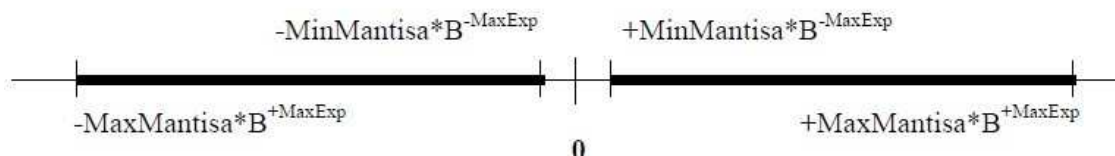
Un número en esta notación tiene infinitas representaciones, de las que se toma la denominada exponencial normalizada, que consiste en que la mantisa no tiene parte entera y el primer dígito a la derecha de la coma es significativo (distinto de cero), salvo en la representación del número 0. El exponente indica donde se sitúa la coma decimal, por eso se habla de coma flotante.

Ejemplos:

a) La representación del número 844,4 (base 10) en notación exponencial normalizada sería $0,8444 * 10^3$

b) La representación del número 0,0000672 (base 10) en notación exponencial normalizada sería $0,672 * 10^{-4}$

El rango de representación en este tipo de formato variará de la siguiente forma:



El cero se representa por todos ceros.

El rango para los números negativos irá desde un número negativo muy pequeño que se aproxime al cero: $-0,000000.....1$ hasta un número negativo muy grande: $-9999...999$

El rango para los números positivos irá desde un número positivo muy pequeño que se aproxime al cero: $+0,000000.....1$ hasta un número positivo muy grande: $+9999...999\ 0$

De esta forma, con un número dado de dígitos representativos se obtiene mayor precisión que con la coma fija, debido a que el valor de estos dígitos es siempre significativo sea el que sea el orden de magnitud del número a representar. Debido a esta adaptación, permite representar un rango mucho mayor de números (determinado por los valores límite que puede tomar el exponente).

Su uso es especialmente interesante en la informática pues permite trabajar con números decimales en rangos amplios, aunque también se usa el truncado de decimales.

La forma más común de usar puntos flotantes es como lo dicta la norma IEEE 754.

Un número en coma flotante consta de 3 partes:

Signo: Expresa el signo del número.

0 ☐ Para números positivos

1 ☐ Para números negativos

Exponente: Representa la orden de magnitud del significando y esta expresado en exceso a $2^{N-1}-1$, ya que el exponente puede ser positivo y negativo.

Mantisa: mantisa o significando, dígitos significativos del número. El tamaño máximo de este campo, usualmente fijo y limitado, determina la precisión de la representación. Este campo está normalizado, es decir, su parte entera sólo consta de un dígito (que será la primera cifra significativa del número a representar y está oculto ya que en binario siempre es 1).

1.1.1. Simple precisión (4 bytes, 32 bits).

Signo	Exponente	Mantisa
1	8	23

Desplazado +127

RANGO: $3.4e^{-038} \dots 3.4e^{+038}$ ERROR DE TRUNCAMIENTO: $2^{1-t} = 0.000000238 \% = 238 \times 10^{-9}$

t : Dígitos para la mantisa

EJEMPLO 1: Codificar en coma flotante de simple precisión el número +12,25. (5 pasos)

1°. Convertir a Binario: 12,25 = 1100,01 2°.

Normalizar: 1100,01 = 1,10001 * 2^{+3} 3°. Signo

positivo, valor 0.

4°. Codificar el Exponente en exceso $2^N - 1$

(según nuestro formato N=8, ya que, de los 4 bytes utilizados, uno es para almacenar el exponente), así pues, el valor del exceso para nuestro formato siempre será 128.

 $127 + (+3) = 130 = 1000\ 0010$

5°. Codificar Mantisa con el uno oculto.

La mantisa es 10001.

La representación es:

Signo (1 bit)	Exponente (8 bits)	Mantisa (23 bits)
0	1000 0010	10001000000000000000000

01000001 01000100 00000000 00000000

EJEMPLO 2: Codificar en formato de coma flotante de simple precisión el valor -324,8125.

1°. Convertir a Binario: -324,8125 = -101000100,1101 (recordar que es negativo) 2°.

Normalizar: -1,010001001101 * 2^8 3°. Signo negativo, valor 1.4°. Codificar el Exponente en exceso $2^N - 1 \rightarrow 127 + (+8) = 135 = 1000\ 0111$ 5°. La

mantisa es 010001001101.

Recordar que hay que añadir el bit de signo 1 a la izda. y completar los 23 bits con ceros por la derecha. Como en este caso se trata de un número negativo.

La representación es:

Signo (1 bit)	Exponente (8 bits)	Mantisa (23 bits)
1	1000 0111	01000100110100000000000

11000011 10100010 01101000 00000000

1.1.2. Doble precisión (8Bytes, 64 bits).

Signo	Exponente	Mantisa
1	11	52

Desplazado +1023

RANGO: $1.7e^{-308} \dots 1.7e^{+308}$ ERROR DE TRUNCAMIENTO: $2^{1-t} = 4,4408 \times 10^{-16} \% t :$

Dígitos para la mantisa

9. CÓDIGOS ALFANUMÉRICOS

Utilizando únicamente ceros y unos, hemos visto como podemos representar en el ordenador información numérica con la que el ordenador (el microprocesador) es capaz de operar con ellos (hacer cálculos aritméticos). Sin embargo, también necesitamos representar otra serie de símbolos (letras mayúsculas y minúsculas; caracteres de puntuación; etc).

Para ello se utilizan los códigos alfanuméricos que, básicamente, se trata de tablas que contienen el conjunto de caracteres que necesitamos representar y a los que se les hace corresponder una combinación de bits siguiendo algún criterio. Esta relación carácter-bits debe ser unívoca, de forma que no haya ambigüedades. Estos códigos alfanuméricos de E/S permitirán traducir la información o los datos que nosotros podemos entender a una representación que la máquina puede interpretar y procesar. Los datos llegan y salen del ordenador a través de los periféricos de Entrada y de Salida, respectivamente. Cada fabricante de componentes de E/S podría asignar una combinación diferente a un mismo carácter o símbolo de origen; sin embargo, esto no sería nada positivo en un mercado como el informático. Por eso se tiende a la estandarización de códigos, que ha llevado a la universalización de unos pocos códigos de E/S, como el EBCDIC, ASCII y Unicode. Antes de seguir aclararemos algunos términos o definiciones:

- **Carácter.**
Componente más pequeño de un lenguaje escrito y que además posee un valor semántico (nota: la semántica estudia el significado de los signos lingüísticos). Ejemplos:
“1”, “.”, “A” “純”.
- **Conjunto de caracteres.**
Es un grupo de caracteres sin ningún valor numérico asociado. Ejemplos: el alfabeto Español o el Cirílico (usado en Rusia y Bulgaria).
- **Codificación de caracteres. (Character encoding)**
Procedimiento por el que asociamos un determinado carácter o símbolo con un valor escalar (número) concreto. En este contexto, a dichos números se les denomina código (en Unicode reciben el nombre de Punto de Código – Code Point-) y deben ser: o Números enteros o No negativos o Estar dentro de un conjunto finito.
- **Conjunto de caracteres codificados.**
Es un grupo de caracteres asociados a un valor escalar. Ejemplo: la letra “A” mayúscula, que según la tabla ASCII tiene el valor decimal 65 (hexadecimal 41, binario 0100 0001).

9.1. Evolución histórica: ASCII, EBCDIC, UNICODE.

A lo largo de la historia de la informática, se han ido utilizando distintos "Sistemas de codificación alfanumérica", los cuales se han distinguido por el número de bits que utilizaban y por la disposición de los caracteres a representar.

La evolución de los distintos sistemas de codificación alfanumérica ha venido determinada por distintos "contextos de época":

1ª Época (Antes de 1960) BAUDOT, FIELDATA, BCDIC El contexto de esta época es el siguiente:

- Pocos ordenadores en el mundo, los más destacados ubicados en EE.UU.
- Eran entornos cerrados y todos de habla inglesa.
- Así pues, cada ordenador podía tener su propio sistema de codificación. Códigos de 5, 6 bits (Baudot, Fieldata, BCDIC, etc)

FIELDATA De 6 bits. Sólo podía representar un conjunto de 26 caracteres o códigos, esto es, 64 en total.

En esos tiempos, el BYTE era igual a 6 bits y no 8.

BCDIC De 6 bits (Binary Coded Decimal Interchange Code). Realmente es un código basado en BCD y utilizado en los equipos de IBM.

2ª Época (Años 1960-70) US-ASCII, ISO 646, EBCDIC El contexto de esta época es el siguiente:

- Surge la necesidad de intercambiar información entre los ordenadores anteriores.

- Todavía es muy limitado el intercambio con ordenadores de otros países y lenguas.
- Se necesita una codificación estándar para el intercambio de información entre dichos ordenadores, así surgen los códigos de 7 bits: US-ASCII, ISO 646 y EBCDIC .USASCII (7 bits + 1 bit de control de paridad)

US-ASCII (7 bits + 1 bit de control de paridad)

*	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	TAB	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	~	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

En esta 1ª versión de ASCII de 7 bits, se definieron 128 códigos (2^7) del 0 al 127. Estos códigos han permanecido fijos en las sucesivas modificaciones posteriores del estándar.

De los 8 bits que se transmitían, sólo se emplearon los de menor peso para establecer los códigos y el de mayor peso se reservó para control de transmisión (paridad): Pxxx xxxx.

Por tanto, esos 128 códigos empiezan por cero en el actual código ASCII de 8 bits.

Los conjuntos de caracteres codificado los dividieron en 4 grupos y sólo contenían los caracteres alfabéticos y símbolos utilizados por los americanos

Grupo 1 (00-1F) Códigos de control, no imprimibles. El 00 es el código nulo.

Grupo 2 (20-3F) Signos especiales, de puntuación y dígitos numéricos. Notar que el espacio en blanco es el 20 (32(10).

Grupo 3 (40-5F) Básicamente las letras MAYÚSCULAS. Notar que no está la Ñ.

Grupo 4 (60-7F) Básicamente las letras minúsculas. Notar que no está la ñ.

Los códigos de control tienen asignadas unas siglas que son las iniciales inglesas de la función a que han sido atribuidas por el organismo de estandarización europeo CCITT (Comité Consultivo Internacional de Telecomunicaciones y Telefonía). Algunos de estos caracteres son:

ACK (Acknowledge)	<u>Acuse de recepción</u> : Carácter de transmisión que es emitido por el dispositivo receptor para indicar al emisor que los datos han sido bien recibidos.
BEL (Bell)	<u>Campana</u> : Carácter utilizado cuando se tiene necesidad de llamar la atención del operador del terminal.
NUL (Null)	<u>Nulo</u> : Carácter nulo que puede estar incluido en un conjunto de caracteres y que puede ser olvidado sin que la información sea efectuada.
SP (space)	<u>Espacio</u> : Carácter blanco que no produce impresión. Se utiliza para separar textos.
VT (Vertical Tabulation)	<u>Tabulación vertical</u> : Produce el que la cabez de impresión vuelva sobre el anterior carácter impreso.
BS (BackSpace)	<u>Vuelta a posición anterior</u> : Produce el que la cabeza de impresión vuelva sobre el anterior carácter impreso.
CAN (Cancel)	<u>Cancelación</u> : Los caracteres a los que va asociado no deben tomarse en cuenta.
CR (Carriage Return)	<u>Retorno de carro</u> : El carro vuelve a la primera posición de la línea.
LF (Line Feed)	<u>Salto de línea</u> : El carro salta una línea, posicionándose el cursor al inicio de ella.
FF (Form Feed)	<u>Salto de primera línea</u> : El carro salta una línea, posicionándose el cursor al inicio de ella.
SYN (Synchronous Idle)	<u>Sincronización</u> : Carácter de sincronización, utilizado para que el receptor se ponga a la escucha y sincronice su reloj.
STX (Start of Text)	<u>Principio de Texto</u> : Precede a una secuencia de caractere, denominada texto y son datos.
ETX (End of Text)	<u>Fin de Texto</u> : Finaliza un texto que va precedido por STX.
ENQ (Enquiry)	<u>Petición</u> : petición de respuesta ala identificación de una estación o al estado.
EOI (End of Transmission)	<u>Fin de comunicación</u> : Indica el fin de la transmisión de uno o varios textos.

ISO 646 Estandarización del US-ASCII de 7 bits. Dado que EE.UU. fue el principal vendedor de computadoras en todo el mundo en aquellos tiempos, el código USASCII fue asumido por la mayoría de los fabricantes como el código estándar internacional (sólo IBM se resistió). Así pues, se hizo necesaria la adaptación de dicho código a otros lenguajes provenientes del latín, en particular los lenguajes del Este de Europa. De este trabajo se encargo la International Organization for Standardization (ISO) dando como resultado la Recomendación ISO 646 en 1967. Se trata de un código ASCII básico excepto en 10 posiciones de carácter que se dejaron abiertas para las "variantes nacionales".

EBCDIC (Extended BCD for Information Communication), código empleado por IBM en sus equipos (va por libre, es el líder del mercado). Lo estrena en sus sistemas S/360 en 1964. La idea básica consistía en anteponer al BCD una zona de 4 bits que se mantienen a 1111 para los dígitos decimales y toman combinaciones distintas para el resto del alfabeto.

3ª Época (Años 1980 - 90) ASCII-8, ISO 8859 El contexto de esta época es el siguiente:

- Con la llegada de los IBM-PC llegó también la necesidad de ampliar el conjunto de caracteres ASCII dado que los ordenadores se extendieron por todo el mundo y no todo el mundo hablaba inglés.
- Es importante adaptar los códigos a todas las lenguas, para que puedan utilizar sus propios caracteres.
- El código ASCII-8 (o simplemente ASCII) se impone como estándar. Se utilizan los 8 bits para representar los códigos de caracteres obviando el bit de paridad.
- Las nuevas normalizaciones aparecen bajo las especificaciones ISO 8859. En ellas, los primeros 128 códigos son los del US-ASCII/ISO 646, y los nuevos 128 códigos disponibles al utilizar el octavo bit, se establecen como sigue:

Part 1: Latin alphabet No. 1 (Revised 1998) ISO 8859-1 Character sets of Western European languages (this is the famous "Latin-1")

Part 2: Latin alphabet No. 2 ISO 8859-2

Character sets of Eastern European languages (Slavic, Albanian, Hungarian, Romanian)

Part 3: Latin alphabet No. 3 ISO 8859-3

Character sets of Southern European languages (Maltese) plus Esperanto

Part 4: Latin alphabet No. 4 (1998) ISO 8859-4 Northern European languages

Part 5: Latin/Cyrillic alphabet ISO 8859-5
 Part 6: Latin/Arabic alphabet ISO 8859-6
 Part 7: Latin/Greek alphabet ISO 8859-7
 Part 8: Latin/Hebrew alphabet ISO 8859-8
 Part 9: Latin alphabet No. 5 ISO 8859-9
 Latin character set used for modern Turkish
 Part 10: Latin alphabet No. 6 (1998) ISO 8859-10)Icelandic, Nordic, and Baltic character sets
 Part 13 (DIS) Latin alphabet No. 7 ISO 8859-13
 Part 14 (DIS) Latin alphabet No. 8 (Celtic) ISO 8859-14

ISO 8859-1 (Latin 1) define la codificación del alfabeto latino, incluyendo los diacríticos (como letras acentuadas Ñ, Ç) y letras especiales ((ß, Ø) necesarios para las lenguas:

afrikaans alemán aragonés catalán, danés	escocés español feroés finés francés	gaélico gallego inglés islandés italiano	neerlandés noruego portugués sueco y vasco.
--	---	--	---

4ª Época (Años 1990 en adelante) UNICODE/UTF-8 El contexto de esta época es el siguiente:

- Se populariza Internet y todos los ordenadores del mundo están intercomunicados. □ Problemas. Buena parte de los codificadores de caracteres tradicionales comparten un problema común: permite procesamientos informáticos bilingües (generalmente usando caracteres latinos y del idioma común), pero no multilingües (procesamiento informático de idiomas arbitrarios mezclados entre ellos).

- Soluciones:

- o Una posible solución, nada práctica, sería tener instalados en nuestros sistemas todos los sistemas de codificación para asegurar el correcto intercambio de información.
- o Una solución más definitiva sería establecer un estándar que reúna todos los caracteres de todos los alfabetos del mundo de forma que todos los ordenadores sean capaces de representar dichos caracteres cargando automáticamente los códigos que sean necesarios.

□ A finales de los años 80, dos organismos estaban desarrollando de forma paralela un Código de Caracteres Unificado:

o Proyecto ISO 10646 De ISO (Organización Internacional de Estandarización) o Proyecto UNICODE Organizado por un consorcio de fabricantes de software.

Cuando se dan cuenta que están trabajando sobre lo mismo deciden unir sus fuerzas pero deciden hacer sus publicaciones por separado. Así pues existe una relación entre ambas publicaciones:

1991 Unicode 1.0 (en ese año se publicó el primer estándar Unicode

(ISBN 0321185781)

1993 Unicode 1.1 es el ISO 10646-1:1993

1996 Unicode 2.0

1998 Unicode 2.1

1999 Unicode 3.0 Cubría 16 bits de la Plana Básica Multilingüe del ISO

10646-1:2000

2001 Unicode 3.1 Introduce Planas suplementarias procedentes del ISO

10646-2:2001

2002 Unicode 3.2

2003 Unicode 4.0 es el ISO 10646:2003

2005 Unicode 4.1

2006 Unicode 5.0

2008 Unicode 5.1.0

2009 Unicode 5.2.0 (Diciembre de 2009)

Unicode 11 es, en estos momentos (Julio 2018).

9.2. Código UNICODE

El objetivo de estas nuevas codificaciones (Unicode e ISO/IEC 10646) es representar cada elemento usado en la escritura de cualquier idioma del planeta. Los idiomas actuales más importantes del mundo ya forman parte de Unicode, incluyendo su puntuación, símbolos especiales, símbolos matemáticos y técnicos, formas geométricas, caracteres gráficos y modelos de Braille.

Unicode proporciona un número único para cada carácter, sin importar la plataforma, sin importar el programa, sin importar el idioma.

Líderes de la industria tales como Apple, HP, IBM, JustSystem, Microsoft, Oracle, SAP, Sun, SyBase, Unisys y muchos otros han adoptado la norma Unicode. Unicode es un requisito para los estándares modernos tales como XML, Java, ECMAScript (JavaScript), LDAP, CORBA 3.0, WML, etc, y es la manera oficial de aplicar la norma ISO/IEC 10646.

Es compatible con numerosos sistemas operativos, con todos los exploradores actuales y con muchos otros productos. La aparición de la norma Unicode y la disponibilidad de herramientas que la respaldan se encuentran entre las más recientes e importantes tendencias en tecnologías de software.

La incorporación de Unicode en sitios web y en aplicaciones de cliente-servidor o de múltiples niveles permite disminuir ostensiblemente los costos del uso de juegos de caracteres heredados. Unicode permite que un producto de software o sitio web específico se oriente a múltiples plataformas, idiomas y países, sin necesidad de rediseñarlo. Además, permite que los datos se trasladen a través de gran cantidad de sistemas distintos sin sufrir daños.

Una idea importante que tenemos que tener en la cabeza para entender estos nuevos códigos es que la solución ha venido de separar la norma de codificación (elección de símbolos y asignación de un código fijo a cada símbolo) de la norma de transmisión (UTF, UCS), usándose ambas en conjunto.

Unicode es, desde 1991, la norma de codificación más universal en la actualidad. Es una gran tabla que, ya en el año 2006, asignaba un punto de código a más de 90.000 símbolos cubriendo la mayor parte de las escrituras usadas actualmente. Investiga cuántos puntos de código cubre actualmente.

En cuando a la norma de transmisión de puntos de código Unicode, se definieron dos métodos de “mapeo” o de localización de caracteres