

Universidad Tecnológica Nacional

Facultad Regional San Nicolás

Introducción a la informática Concepto de Algoritmos Tratamiento de datos. Parte 1

Informática para Ingeniería

Ing. Silvia Kern 2008 UTN – FRSN

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

CUESTIONES PARA TENER EN CUENTA:

Vas a encontrar estos signos, te van a ayudar.

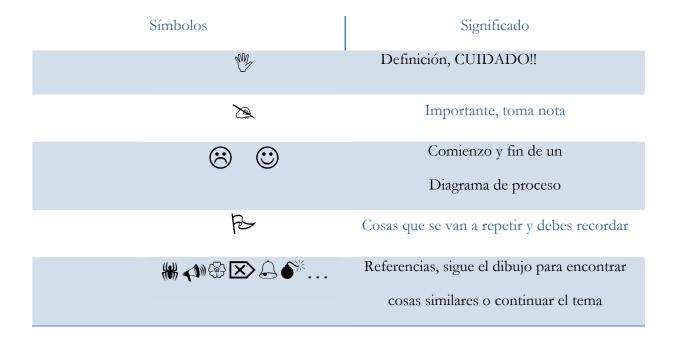
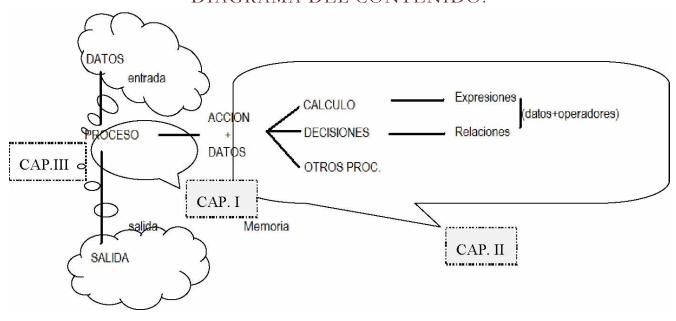


DIAGRAMA DEL CONTENIDO:



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

CAPITULO I. REPASANDO CONCEPTOS

Conceptos básicos y metodología para la solución de problemas por medio de computadoras.

OBJETIVO:

- Conocer la terminología relacionada con los algoritmos; así como la importancia de aplicar técnicas adecuadas de programación.
- Conocer la metodología resolución de problemas en cada una de sus etapas, en este caso revisando los dispositivos disponibles.

INTRODUCCIÓN

La computadora no solamente es una maquina que puede realizar procesos para darnos resultados, sin que tengamos la noción exacta de las operaciones que realiza para llegar a esos resultados. Con la computadora además de lo anterior también podemos diseñar soluciones a la medida, de problemas específicos que se nos presenten. Mas aun, si estos involucran operaciones matemáticas complejas y/o repetitivas, o requieren del manejo de un volumen muy grande de datos.



El diseño de soluciones a la medida de nuestros problemas, requiere como en otras disciplinas una metodología que nos enseñe de manera gradual, la forma de llegar a estas soluciones.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



A las soluciones creadas por computadora se les conoce programas ¹ como y no son más que una serie de operaciones que realiza la computadora para llegar a un resultado, con un grupo de datos específicos. Lo anterior nos lleva al razonamiento de que un programa nos sirve para solucionar un problema específico.

Para poder realizar programas, además de conocer la metodología mencionada, también debemos de conocer las funciones que puede realizar la computadora y las formas en que se pueden manejar los elementos que hay en la misma.

Repasemos conceptos que vamos a utilizar!

Computadora: Es un dispositivo electrónico utilizado para procesar información y obtener resultados. Los datos y la información se pueden introducir en la computadora como entrada (input) y a continuación se procesan para producir una salida (output).

¹ Programa: secuencia de órdenes que cumple una computadora.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



PROCESO DE INFORMACIÓN EN LA COMPUTADORA



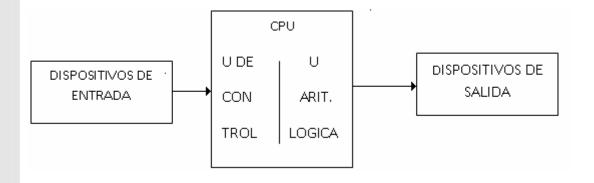




Programa: Es un conjunto de instrucciones escritas de algún lenguaje de programación y que ejecutadas secuencialmente resuelven un problema especifico.



ORGANIZACIÓN FÍSICA DE UNA COMPUTADORA



Este diagrama suele llamarse "de arquitectura interna"

Dispositivos de Entrada: Como su nombre lo indica, sirven para introducir datos en la computadora para su proceso. Los datos se leen desde los dispositivos de entrada y se almacenan en la memoria central o interna. Ejemplos: teclado, scanners (digitalizadores de rastreo), mouse (ratón), trackball (bola de ratón estacionario), joystick



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

(palancas de juego), lápiz óptico.



Dispositivos de Salida: Regresan los datos procesados que sirven de información al usuario. Ejemplo: monitor, impresora

La Unidad Central de Procesamiento (CPU) se divide en dos:

- Unidad de control
- Unidad Aritmético Lógica
- ➤ Unidad de Control: Coordina las actividades de la computadora y determina que operaciones se deben realizar y en que orden; así mismo controla todo el proceso y ordenes a ejecutar coordinando las respuestas de los dispositivos.



Unidad Aritmético - Lógica: Realiza operaciones aritméticas y lógicas, tales como suma, resta, multiplicación, división y comparaciones.

La Memoria de la computadora guarda datos y secuencias, presenta dos formatos a utilizar:

- Memoria Central o Interna
- Memoria Auxiliar o Externa

Memoria Central (interna): La CPU utiliza la memoria de la computadora para guardar información mientras trabaja con ella; mientras esta información permanezca en memoria, la computadora puede tener acceso a ella en forma directa. Esta memoria construida internamente se llama memoria de acceso aleatorio (RAM).

La memoria interna consta de dos áreas :

La memoria RAM (Random Access Memory): Recibe el nombre de memoria principal o memoria de trabajo, en ella se almacena información solo mientras la



almacenan procesos y variables!

CUIDADO!

Aquí se

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

computadora esta encendida. Cuando se apaga o arranca nuevamente la computadora, la información se pierde, por lo que se dice que la memoria RAM es una memoria volátil.

La memoria **ROM** (**Read Only Memory**): Es una memoria estática que no puede cambiar fácilmente, la computadora puede leer los datos almacenados en la memoria ROM, pero no se pueden introducir datos en ella, o cambiar los datos que ahí se encuentran; por lo que se dice que esta memoria es de solo lectura. Los datos de la memoria ROM están grabados en forma permanente y son introducidos por el fabricante de la computadora, de acuerdo a la conformación de que dispone.

Memoria Auxiliar (Externa): Es donde se almacenan todos los programas o datos que el usuario desee. Los dispositivos de almacenamiento o memorias auxiliares (externas o secundarias) más comúnmente utilizados son: cintas magnéticas y discos magnéticos.





PREGUNTITAS DE ATENCIÓN:

Cuando tu PC arranca, se encienden varias luces alternativamente, indicando que se está controlando algún dispositivo.

Por favor, revisa y anota esa secuencia, prestando atención a las comprobaciones que dejan datos en la pantalla (disco rígido, memoria, CDs, etc.)

Ahora apaga tu PC (utilizando el orden normal de apagado) y enciéndela con algún dispositivo desconectado, p.e. tu teclado.

Presta especial atención ahora a la pantalla... han cambiado los mensajes?...

Cuando los mensajes están asociados a los dispositivos y su funcionamiento suelen hacer referencia a ellos en palabras del idioma ingles. Podrías reconocer algunos de esos términos y a qué hacen alusión?...

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

LENGUAJE Y COMUNICACIÓN





Lenguaje: Es un conjunto de símbolos que sirven para transmitir uno o mas mensajes (ideas) entre dos entidades diferentes. A la transmisión de mensajes se le conoce comúnmente como comunicación.

La *comunicación* es un proceso complejo que requiere una serie de reglas simples, pero indispensables para poderse llevar a cabo. Las dos principales son las siguientes:

- Los mensajes deben correr en un sentido a la vez.
- Deben forzosamente existir 4 elementos: Emisor, Receptor, Medio de Comunicación y Mensaje.

LENGUAJES DE PROGRAMACIÓN

Cuando un lenguaje esta adaptado para comunicar ordenes a un ordenador, se llama "de programación". Consta de un conjunto de símbolos, caracteres y reglas sintácticas, que le permiten enviar ordenes (programas) y recibir respuestas, para que las personas se comuniquen con la computadora, entendiendo las respuestas.



Los lenguajes de programación tienen un conjunto de instrucciones que nos permiten realizar operaciones de entrada/salida, cálculo, manipulación de textos, lógica/comparación y almacenamiento/recuperación.

Los lenguajes² de programación se clasifican en:

> Lenguaje Máquina: Son aquellos cuyas instrucciones

Nosotros no podemos

son directamente entendibles por la computadora y no necesitan

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

traducción posterior para que la CPU pueda comprender y ejecutar el programa. Las instrucciones en lenguaje maquina se expresan en términos de la unidad de memoria mas pequeña el bit (dígito binario 0 o 1).

leerlo ¡!

➤ Lenguaje de Bajo Nivel (Ensamblador): En este lenguaje las instrucciones se escriben en códigos alfabéticos conocidos como mnemotécnicos para las operaciones y direcciones simbólicas, es decir son pocas abreviaturas de palabras pertenecientes a lenguaje humano que conjuntamente con señales ubicadas en determinados espacios permiten escribir órdenes.



Lenguaje de Alto Nivel: Los lenguajes de programación de alto nivel (BASIC, pascal, cobol, fortran, etc.) son aquellos en los que las instrucciones o sentencias a la computadora son escritas con palabras similares a los lenguajes humanos (en general en ingles), lo que facilita la escritura y comprensión del programa.

Escribimos sobre un editor de textos básicos como el pad.

En general, los lenguajes de programación son catalogados como de "alto nivel" (los que son muy parecidos al lenguaje humano) y de "bajo nivel" (aquellos cuyo formato se asemeja más al lenguaje máquina).

Un algoritmo, cuando necesita ser ejecutado en un ordenador, se escribe con el formato del lenguaje de programación elegido respetando sus reglas sintácticas en un editor particular, (pero puede escribirse en cualquier editor básico, notepad por ejemplo), y se guarda con la extensión correspondiente.

Este programa se denomina "fuente³" (está escrito en palabras que reconocemos, y es el origen de todo el proceso). Pero la secuencia de ordenes no puede ser ejecutado

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

porque el computador no conoce el formato en que se expresan las órdenes, para poder ser reconocido, se debe rescribir el programa en "lenguaje objeto" que tiene ordenes entendibles por el procesador (escrito en ceros y unos, en muy alto nivel).

M

Para esto, el paquete del lenguaje correspondiente, lo traduce (por medio de un "traductor" o lo "compila" (puede tener una o ambas de las herramientas). Un *traductor* "traduce" a lenguaje máquina cada una de las instrucciones (en memoria) y las ejecuta, cortando el proceso cuando encuentra algún error (vuelve a la fuente, se arregla y se traduce nuevamente).

El "compilador" tiene un proceso más completo y complejo. En primer lugar se ocupa de los errores y los marca en pantalla de diferentes maneras, uno a uno. Cuando el proceso completo está libre de errores, procede a enlazar las rutinas (pedazos que se encadenan al proceso general) que componen el programa, con rutinas de librería propias, matemáticas, de uso de dispositivos, etc. o del usuario; y finalmente, produce una salida en lenguaje objeto que es ejecutada por el procesador (o almacenada en disco).

Los errores que analiza y marca un compilador se relacionan con la gramática y la sintaxis del lenguaje: es decir, si alguna palabra está escrita en forma incorrecta, si todos los lazos están cerrados, si las variables han sido declaradas o utilizadas en forma correcta (uso de la memoria), si existen las rutinas que se citan, etc. Sin embargo, no puede descubrir errores de lógica, es decir *un programa bien compilado no necesariamente resuelve el problema planteado*, solamente está escrito en forma entendible para ser procesado.





Preguntitas de atención:

Supongamos que identificas un programa fuente, en tu PC... con qué utilitario lo abrirías para no dañarlo? Como lo guardarías?

Y si fuera un programa ejecutable... como lo reconoces o diferencias??

Te ha pasado alguna vez, que al ejecutar un programa te aparece un cartel diciendo que falta alguna DLL? Ahora has encontrado la razón y tienes idea de a qué se refiere ese aviso? Cómo lo solucionarías?

Introducción a

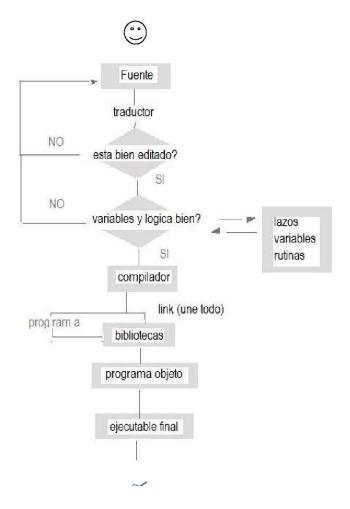
Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

DIAGRAMA DEL PROCESO

De escritura de programa fuente y generación de objeto y ejecutable



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



Y QUÉ ES UN ALGORITMO...?

La palabra algoritmo se deriva de la traducción al latín de la palabra árabe "alkhowarizmi", nombre de un matemático y astrónomo árabe que escribió un tratado sobre manipulación de números y ecuaciones en el siglo IX.





Un algoritmo es una serie de pasos finitos, definidos y organizados que describe el proceso que se debe seguir, para dar solución a un problema especifico

EJECUTAR o probar un algoritmo es realizar las tareas que fijan esos pasos, en el orden especificado y utilizando los recursos

CARACTERÍSTICAS DE UN ALGORITMO

Las características que debe poseer una secuencia de tareas para considerarse algoritmo son: precisión, definitud y finitud.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Precisión

Definitud o determinismo

Finitud

El algoritmo debe indicar el orden exacto de ejecución de cada tarea.

Si se sigue el algoritmo dos o más veces con los mismos datos de entrada, se deben obtener los mismos datos de salida.

Si se utiliza un conjunto de datos análogos se obtienen resultados comparables y con la misma seguridad en su valor El algoritmo debe terminar en algún momento y debe usar una cantidad de recursos finita.

Cuando se implementa un algoritmo en un computador digital, los recursos con los que se cuenta son *tiempo de proceso y memoria*.

Siempre es necesario repasar el proceso que escribimos para resolver un problema, para hacer que sea *económico* es decir, que observe la economía de tiempo y de memoria para evitar perdidas derivadas del mal uso de ellas. Un proceso es *eficiente* cuando tiene la menor cantidad de pasos posibles, y el mejor uso de memoria (variables) para que, al ejecutarlo se obtenga la solución buscada.

Hay algunas formas de escribir los algoritmos y facilitarnos luego la traducción a lenguajes de programación. Existen serie de símbolos y reglas que se utilizan para describir de manera explícita un proceso. Hay *formatos gráficos*, para producir diagramas de seguimiento de un proceso (pueden ser diagramas de flujo o DFD, o diagramas estructurados, denominados de Chapín) y una forma de escribir las ordenes con palabras especialmente elegidas, que se llama pseudo código.



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

COMO LOS ESCRIBIMOS??...

Para poder escribir un proceso correcto, siempre tenemos que tener en cuenta los siguientes pasos.

a) Definición del Problema

Esta fase está representada por el enunciado del problema, si lo hay, que debe ser comprendido y delimitado, para lo cual requiere una definición clara y precisa. Es importante que se conozca lo que se desea que realice la computadora con el juego de datos predeterminado; es decir qué esperamos obtener, mientras esto no se conozca del todo no tiene mucho caso continuar con la siguiente etapa. Hay casos en que es necesario acotar el juego de datos, cuando no está especificado y puede anular el proceso posterior. Por ejemplo, si hay una división, el denominador debe acotarse a que sea diferente de cero, aunque el problema no lo especifique, para evitar un error y perder el control.

b) Análisis del Problema

Una vez que se ha comprendido lo que se desea de la computadora, es necesario conocer: cuales son los datos de entrada, cual es la información que se desea producir (salida) y si hay métodos y/o fórmulas necesarios para procesar los datos.

c) Definición los pasos. Después procederemos a definir los pasos que creemos necesarios para resolver el problema, en forma ordenada, y precisa.

Las características de un buen proceso de solución son:

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

- Debe haber un punto particular de inicio.
- Debe escribirse en pasos simples y de resolución única.
- El proceso completo tiene que ser definido, no debe permitir dobles interpretaciones ni ambigüedades.
- Debe ser general, es decir, soportar la mayor cantidad de variantes que se puedan presentar en la definición del problema.
 - Debe ser finito (limitado) en tamaño y tiempo de ejecución.
 - Debe poder arribarse a la solución solicitada.

Una vez conseguido el proceso, hemos armado nuestro algoritmo, y deberemos escribirlo en un lenguaje de programación para enviar las ordenes a un procesador.

Los procesos pueden escribirse de diferentes formas facilitando nuestra tarea. Una forma es un lenguaje similar al nuestro que se llama Pseudocódigo y usa algunas palabras especiales (escribir, ingresar, mostrar, etc.). También se usan gráficos como los llamados diagramas de Chapin, o los diagramas de Flujo que marcan el sentido en que se desarrolla el proceso.

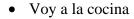
En nuestro caso, podremos utilizar una herramienta denominada DFD que nos permite pasar estos diagramas a un proceso que los prueba tanto en su ejecución como en el tratamiento de sus variables y salidas.

Importante

- ❖ Analizar el entorno
- Limitar el problema
- Definir pasos
- ❖ Verificar datos

DIAGRAMA es un dibujo que marca las acciones, variables y sentido del proceso. DFD es un utilitario para probar los procesos sin escribir los programas en ningún lenguaje de programación

Por ejemplo: Pasos que cumplo para servirme un vaso de agua de la heladera.



- busco un vaso
- abro la heladera
- saco la jarra de agua fría
- sirvo agua en el vaso
- coloco la jarra en su lugar
- cierro la heladera.
- listo

Revisemos: tengo una "variable" que cambiará: el vaso que tiene un determinado volumen y está vacío, luego será llenado (poco, medio, mucho, no importa); hay otra que es la jarra de agua, que tiene un contenido inicial y una ubicación (en la heladera); podemos reconocer también una "operación" que involucra a dos "variables": sirvo agua en el vaso, cambiando "contenido" de un



lugar a otro; tenemos un par de instrucciones que deben aparecer juntos: abrir la heladera y cerrarla.

Y finalmente el proceso está cumplido, con lo cual termina la secuencia de acciones.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Preguntitas de atención:

Escribirías en una secuencia las acciones que ejecutas para ponerte las zapatillas??

FIN DEL CAPÍTULO 1: ACTIVIDADES DE FINAL DEL CAPÍTULO

Revisemos nuestros objetivos

Recuerda que estas preguntas son para pensar, si alguna no sabes como resolverla, simplemente vuelve a leer estas paginas. No tienes que presentarlas al docente, ni te darán nota por ellas son parte de una evaluación propia que debes hacer al final de tu lectura... solo se sincero...

Nro	Pregunta	Es Verdad?
1	La compu es un dispositivo para resolver problemas	
2	Un programa es un conjunto de instrucciones para resolver un algoritmo	
3	Los dispositivos de entrada/ salida son los mismos	
4	La memoria central guarda solo datos, la memoria auxiliar guarda los procesos	
5	El lenguaje maquina es similar al lenguaje humano	
6	Un compilador es más completo que un traductor	
7	Si el programa escrito en un lenguaje es compilado completamente, seguramente resuelve el problema.	
8	Un algoritmo es como una receta de cocina : muestra los pasos necesarios para solucionar un problema.	
9	Si un algoritmo no es eficiente, puede servir igual para resolver el problema	

Recuperamos ideas:

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Revisá las definiciones de lenguaje y los tipos

Mirá bien la definición de algoritmo

• Acordate que un buen algoritmo siempre tiene que tener :

Inicio y final

Pasos simples

Solución única

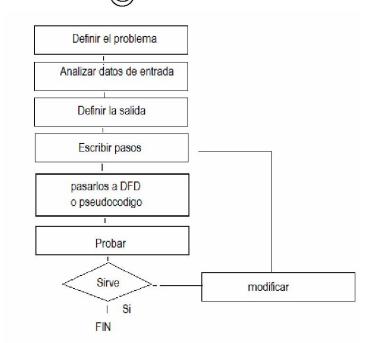
Utilidad y eficiencia.

Glosario del tema:



Programa Lenguaje de programación Programa fuente Programa Objeto

Algoritmo Diagrama de Flujo DFD

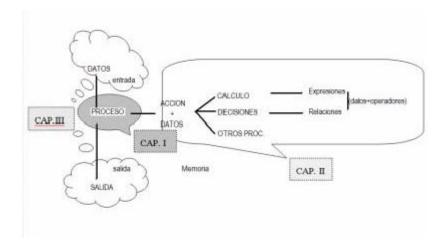


Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

En el proceso de aprendizaje de la primera pagina... Resumimos : Lo marcado en gris te indica hasta donde hemos avanzado!!



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

CAPITULO II. ELEMENTOS BÁSICOS

Entidades primitivas para el desarrollo de algoritmos

Tipos de datos

Expresiones

Operadores y operandos

Identificadores como localidades de memoria

OBJETIVO:

- Crear espacios de memoria para almacenar datos de los problemas y definirlos correctamente.
- Conocer las reglas para escribir formulas matemáticas a expresiones validas para la computadora.
- Diferenciar constantes e identificadores y tipos de datos simples.

Símbolos	Significado
M.	Definición, CUIDADO!!
	Importante, toma nota
⊗ ©	Comienzo y fin de un Diagrama de proceso
R	Cosas que se van a repetir y debes recordar
₩ ••• 🖾 🖾 ••	Referencias, sigue el dibujo para encontrar cosas similares o continuar el tema

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

TRATAMIENTO DE DATOS

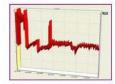
LOS DATOS

Cuando enfrentamos un problema tenemos algunas magnitudes que se han medido o recopilado, que nos sirven para representar cosas en el proceso de resolución con alguna relación entre ellos o entre valores que toman en diferentes momentos del proceso. Pueden tener valores diferentes o similares, de texto o numéricos, lo que nos interesa es que deben ser almacenados para poder trabajar con ellos. Y tenemos que diferenciar las operaciones que podemos hacer con ellos según su naturaleza. Veremos entonces los "tipos de datos".

TIPOS DE DATOS



Todos los datos tienen un tipo asociado con ellos. Un dato puede ser un simple carácter, tal como 'b', un valor entero tal como 35. El tipo de dato determina la naturaleza del conjunto de valores que puede almacenarse en un lugar de la memoria. Solo usaremos datos simples, que son los que tienen un solo valor almacenado a la vez.



TIPOS DE DATOS SIMPLES



- Datos Numéricos: Permiten representar valores escalares de forma numérica, esto incluye a los números enteros y los reales. Este tipo de datos permite realizar operaciones aritméticas comunes.
- Datos Lógicos: Son aquellos que solo pueden tener dos valores (cierto o falso) ya que representan el resultado de una comparación entre otros datos (numéricos o alfanuméricos).



• Datos Alfanuméricos (cadenas): Es un conjunto de caracteres alfanuméricos (letras o

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

números) que permiten representar valores identificables de forma descriptiva, esto incluye nombres de personas, direcciones, etc. Es posible representar números como alfanuméricos, pero: estos pierden su propiedad matemática, es decir no es posible hacer operaciones con ellos.

Ejemplo "Facultad Regional San Nicolás"

Nota según el tipo de datos ocuparan mas o menos lugar en la memoria para almacenarlos y efectuar operaciones con ellos (mirá la tabla al fin del capitulo).

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

· Cuidado!

Verás que cuando nos

referimos a expresiones y operadores NO es lo

mismo que haces en Matemática...

Harías la comparación?

EXPRESIONES



Las expresiones son combinaciones de datos, sean constantes, o variables, y símbolos de operación, paréntesis y nombres de funciones especiales. Por ejemplo:

$$a + (b + 3)/c$$

Cada expresión toma un valor que se determina reemplazando los valores de las variables y constantes, y ejecutando las operaciones indicadas.

Una expresión consta de operadores y operandos.

Según sea el tipo de datos que manipulan, se clasifican las expresiones en:

- -Aritméticas
- -Relacionales
- -Lógicas

OPERADORES Y OPERANDOS

• *Operadores:* Son elementos que relacionan de forma diferente, los valores de uno o mas datos y/o constantes. Es decir, los operadores nos permiten manipular valores.

Tipos de operadores

Aritméticos

Relacionales

Lógicos

OPERADORES ARITMÉTICOS:

Los operadores aritméticos permiten la realización de

Materias Básicas

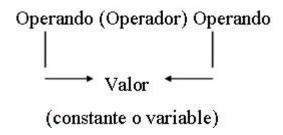
Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



operaciones matemáticas con los valores (variables y constantes).

Los operadores aritméticos pueden ser utilizados con tipos de datos enteros o reales. Si ambos son enteros, el resultado es entero; si alguno de ellos es real, el resultado es real.







Operadores Aritméticos

- + Suma
- Resta
- * Multiplicación
- / División

Mod Modulo (residuo división entera)

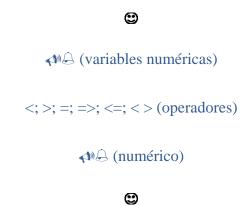
Ejemplos:

Expresion	resultado.
7/2	3.5
12 MOD 7	5
4 + 2*5	14

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Los Operadores Aritméticos Se usan para operaciones aritméticas simples.

Prioridad de los Operadores Aritméticos

> Todas las expresiones entre paréntesis se evalúan primero. Las expresiones con paréntesis anidados se evalúan de dentro a fuera, el paréntesis más interno se evalúa primero.



- > Dentro de una misma expresión los operadores se evalúan en el siguiente orden.
 - 1.- ^ Exponenciación
 - 2.- *, /, mod Multiplicación, división, modulo.
 - 3.- +, Suma y resta.

- Indica en qué orden tengo que aplicarlos para resolver, ó cual va primero...
- Los operadores en una misma expresión con igual nivel prioridad se evalúan de izquierda a derecha.

Ejemplos:

•
$$4 + 2 * 5 = 14$$

$$46 / 5 = 9.2$$

•
$$3 + 5 * (10 - (2 + 4)) = 23$$

$$3 + 5 * (10 - 6) = 3 + 5 * 4 = 3 + 20 = 23$$

$$3.5 + 5.09 - 3.5 = 8.59 - 3.5 = 5.09$$

•
$$2.1 * (1.5 + 3.0 * 4.1) = 28.98$$

$$2.1 * (1.5 + 12.3) = 2.1 * 13.8 = 28.98$$

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



PREGUNTITAS DE ATENCIÓN:

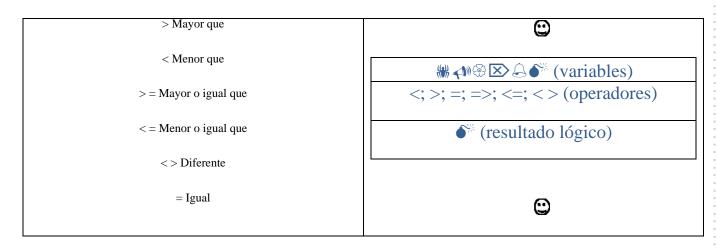
Que puede pasar con los operadores aritméticos aplicados a datos alfabéticos?



OPERADORES DE COMPARACIÓN:

- > Se utilizan para establecer una comparación entre dos valores.
- Compara estos valores entre si y esta comparación produce un resultado verdadero o falso.
- Los operadores de comparación comparan valores del mismo tipo (numéricos o cadenas)
- > Tienen el mismo nivel de prioridad en su evaluación.
- > Tienen menor prioridad que los aritméticos.





Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Ejemplos:

a - b < c Verdadero

a - b = c Falso

a * b < > c Verdadero

Ejemplos no lógicos:

"T" < 30 (no es lógico porque tiene diferentes tipos de datos)

Los operadores de comparación se usan para comparar dos datos o valores, como mayor, menor, o igual.

Sugeridos a resolver:

Evaluar

A) (9+3) * 5 DIV 4 MOD 7 + 2

Rta 2

b) 2.1 * (1.5 + 3.0 * 4.1)

Rta. 28.98

c) 3.5 + 5.09 - 14.0 / 4

Rta. 5.09

d)
$$3 + 4 * (8*(4-(9+3)/6))$$

e)
$$4 - 40 \text{ div } 5$$

f) a div a div a * b

Materias Básicas

Introducción a la Informática -2008

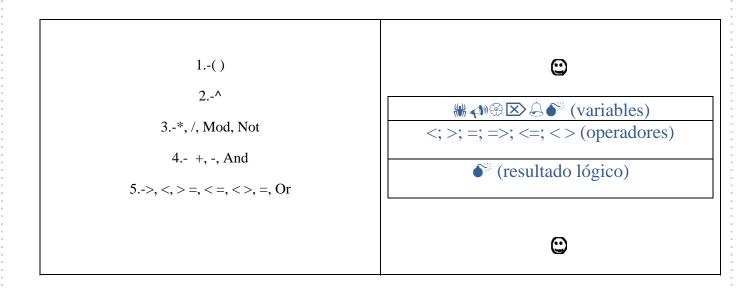
DISEÑO DE ALGORITMOS

OPERADORES DE RELACIÓN:

- > Se utilizan para unir dos o más comparaciones.
- Las Comparaciones se enlazan con los operadores "Y" y "O" y producen un resultado verdadero o falso.
- ➤ EL Operador Y (también AND) es una SUMA e indica que solamente saldrá resultado como verdadero, si ambas comparaciones son verdaderas.
- ➤ El Operador O (también OR) es un PRODUCTO e indica que saldrá resultado como verdadero, si una de comparaciones es verdaderas..
- ➤ Los resultados indicados se expresan en las llamadas TABLAS DE VERDAD.

Operadores de relación: son los que permiten enlazar varias comparaciones.

Prioridad de los Operadores en General



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



Un identificador es un nombre especial que sirve para "identificar" una posición en la memoria de la computadora, que nos permite acceder al contenido, que son los datos del problema.

Ejemplo: Nombre; Num_hrs; Calif2

Para poner nombre a una posición de memoria, o identificarla, conviene que el nombre que pongamos sea fácil de recordar y de utilizar, así que le pondremos nombres que tengan que ver con lo que guardan, y que no sean demasiado largos. Generalmente se utilizan en ellos, letras o una combinación de letras y números, y el símbolo de subrayado, porque los demás símbolos, como ya vimos, se utilizan en las operaciones.



Los identificadores, entonces pueden guardar en la memoria de la comp., valores que sean constantes o que cambien durante el proceso.

"Identificador" es el nombre que le damos al lugar de la memoria que puede guardar un dato. Lleva como propiedad un tipo de dato que puede guardar, y de acuerdo a él adapta su tamaño.



CONSTANTES Y VARIABLES

Ejemplo:

* Constante: Una constante es un dato numérico o alfanumérico que no cambia durante el proceso.

pi = 3.1416

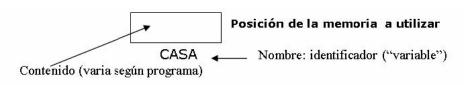
* *Variable:* Es un espacio (en la memoria de la computadora) donde se guarda un dato durante el proceso. Para poder guardar ese dato, necesitamos darle un nombre para después utilizarla. En muchos casos necesitamos además decir cuanto espacio va a necesitar, definiendo el tipo de cosa que vamos a guardar ahí.

Ejemplo:

área = pi*radio ^ 2

Las variables son : el radio, el área y la constate es pi

(veremos el ejemplo en capitulo siguiente)





Preguntitas de atención:

Si dos variables contienen un carácter distinto almacenado cada una, en base a qué puedo decir cual es mayor? Y si fueran palabras almacenadas como cadenas?

ANALICEMOS Y AGREGUEMOS INFORMACIÓN ÚTIL

Una vez que sabemos en qué consiste un algoritmo veremos como nos arreglamos para introducir datos al proceso.

Toda información viene representada por medio de datos, que son los que el algoritmo debe manipular. Es decir, es tan importante especificar las acciones que va a realizar el algoritmo como los datos que éste va a utilizar.

Un algoritmo puede utilizar datos de diferentes clases y cada una de éstas es un *tipo de datos*. Un tipo de datos puede definirse como un conjunto, sobre cuyos elementos se define a su vez un conjunto de operaciones.

Ahora analizaremos como guardar los datos que necesitamos utilizar en un proceso, y las operaciones que podemos hacer con ellos.

Tipo	Operadores aritméticos	Operadores de comparación
ENTEROS	+ -* div mod	<;>;=>;<=;<>
REALES	+ -* /	<;>;=>;<=;<>
CARÁCTER	+	<;>;=>;<=;<>
CADENAS	+	<; >; => ; <=; <>
LOGICOS	Según tablas	<;>;=>;<=;<>

Ahora bien, según el tipo que le demos a las variables, ocupan más o menos lugar de memoria. Por eso es necesario definir de antemano qué se guardará en ellas y con que capacidad necesitamos contar.

Analizando, encontramos que las variables numéricas, ocupan la cantidad de bytes que aparecen a continuación, y pueden contener valores que pertenezcan al rango de números descrito en la tabla.

DISEÑO DE ALGORITMOS

	en bytes	rango desde	hasta
ENTEROS	2	-32767	32767
ENTERO LARGO	4	-2147483647	2147483647
ENTERO SIN SIGNO	2	0	65535
REALES SIMPLES	4	1,6 E-45	3,4 E+38
REALES DOBLE PRECISION (DIEZ DIGITOS)	8	5 E-324	1,1 E+4932

Loa alfabéticos ocupan tantos bytes como caracteres y los lógicas ocupan un lugar mínimo por ser biestables, es decir 1 bit.

Lógico o booleano. Solo dos valores posibles, *cierto* y *falso*, operaciones lógicas: conjunción (y), disyunción (o) y negación (no).

A	В	A y B	A o B	no A
cierto	cierto	Cierto	Cierto	Falso
cierto	falso	Falso	Cierto	Falso
falso	cierto	Falso	Cierto	Cierto
falso	falso	Falso	Falso	Cierto

Tabla 1: Resultados de las operaciones lógicas básicas

En general, el tipo de una variable es numérico (real o entero), carácter ó lógico...



Preguntitas de atención:

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

- Si almacenamos un número real en un espacio entero, daría error?
- Podríamos multiplicar una variable que almacena un carácter por un entero constante?
- Podes especificar las diferencias entre división, división entera y modulo?
- Hablando de precisión... podrías explicar el término... ¿?

FIN DEL CAPÍTULO 2: ACTIVIDADES DE FINAL DEL CAPÍTULO

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Revisemos nuestros objetivos

Nro	Pregunta		Verdad o no?
1	Ordenados según la cantidad de lugar que ocupan, la siguiente secuencia es válida?		
	Enteros – alfabéticos – rea	iles dobles	
2	Los siguientes nombres so	on validos para identificadores?	
	a. O ²		
	bDATO		
	c. OTRO_DA		
	d. M*2		
	e. 2MIO		
	f. M_2		
3	Analiza si cada variable ti	ene bien adjudicado el tipo:	
	a. $A = 248/5$	ENTERO	
	b. $MIS = 58 E+4$	REAL DOBLE	
	e. LETRA = "436"	CADENA DE 5 CARACT	
	d. TUYO = FALSO	CADENA DE 5 CARACT	
	e. TUYO = FALSO	LOGICO	
	f. $CUARTO = 4$	ENTERO	
4		omo constante, a lo largo del proceso no	
	THE RESERVE OF THE PROPERTY OF	do, entonces una constante no puede ser signo igual en una expresión.	
	and and and and		

Recuperamos ideas:

- Revisá las definiciones de constantes y variables
- Mirá bien la definición de los tipos que pueden tener

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

- Compará las expresiones algorítmicas de informática con las que utilizás en matemática.
- Acordate de ver donde y como se aplican los operadores
 - aritméticos
 - de comparación
 - de relación.
- > Revisá el orden en que se deben aplicar.

Glosario del tema:

- Variables
- Operadores
- Memoria
- Comparación
- Relación
- > Tablas de Verdad.

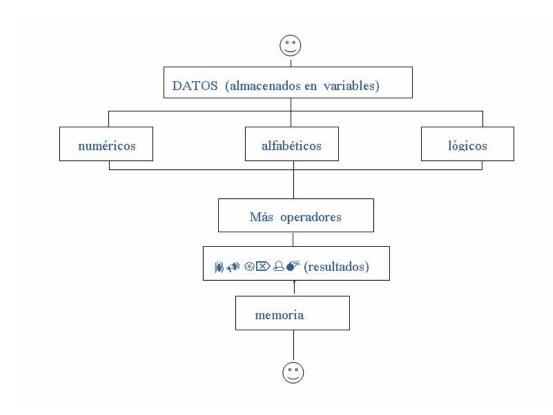
Materias Básicas

Introducción a la Informática -2008

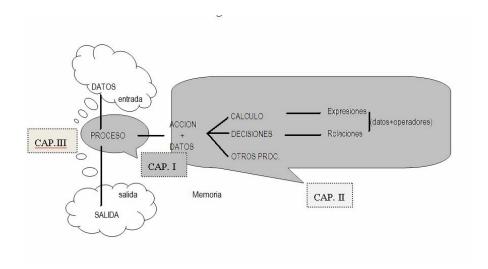
DISEÑO DE ALGORITMOS



En el proceso de aprendizaje de la primera pagina...



Lo marcado en gris te indica hasta donde hemos avanzado!!



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

CAPITULO III. LOS PROCESOS SECUENCIALES

Manipulación de datos en un proceso secuencial simple

Ingreso de datos y almacenamiento Cálculos, expresiones y procesos

Salida de datos

Formatos de las operaciones de i/o

OBJETIVO:

- Reconocer la forma de ingresar datos para ser almacenados y utilizados en un proceso
- Disponer de esos datos correctamente
- Aprender a utilizar operaciones de salida para mostrar los resultados obtenidos en un proceso.
- Incorporar lenguaje técnico apropiado

Símbolos	Significado		
M.	Definición, CUIDADO!!		
	Importante, toma nota		
	Comienzo y fin de un Diagrama de proceso		
	Cosas que se van a repetir y debes recordar		
	Referencias, sigue el dibujo para encontrar cosas similares o continuar el tema		

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

ESTRUCTURAS SECUENCIALES



Un proceso secuencial es la forma que toma un proceso si cada acción (instrucción) sigue a otra ordenadamente. Las tareas se suceden de tal modo que la salida de una es la entrada de la siguiente y así sucesivamente hasta el fin del proceso. La secuencia se representa de la siguiente forma:

Inicio

Accion1

Accion2

•

AccionN

Fin

Generalmente los procesos que se suceden son de tres tipos:

- Asignaciones, para guardar datos en memoria o hacer cálculos,
- lecturas para ingresar datos y
- salidas de información, para mostrar resultado

Veamos las utilidades de cada tipo



-Asignación:

La asignación consiste, en el almacenamiento de valores o resultados a una zona de la memoria. La zona será reconocida con el nombre de un identificador ó "variable" que recibe el valor.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Tienen la particularidad de que siempre se "entienden" o analizan resolviendo las operaciones que se sitúan a la derecha del signo igual, y el valor resultado se almacena en la variable que se nombra a la izquierda del signo.

De acuerdo a lo anterior, se derivan dos condiciones muy importantes:



- -En la primera parte del proceso, las variables que participan en las operaciones del lado derecho se UTILIZAN, es decir, *se lee su valor y se usa en alguna operación*. Con este movimiento el contenido de la variable (el número almacenado) no se modifica, solamente es leído.
- -La variable que aparece a la izquierda del signo igual, es la que *recibe el resultado para ser almacenado*, en la segunda parte del proceso. Por lo tanto, si había algún dato guardado allí, se borra al superponerse el nuevo valor a almacenar.

Y teniendo en cuenta lo anterior, sacamos una conclusión que nos va a servir en todos los procesos...



Podemos utilizar la misma variable para operar con un valor en la parte derecha de la expresión, y cuando está el resultado, almacenarlo en ella misma, con lo que el viejo valor se sustituye por en nuevo sin problemas².

Ejemplo:

TOTAL = 458

DATO= 120

Si hacemos TOTAL = TOTAL + DATO



Vamos por pasos. Supongamos tener guardado en TOTAL el valor 458 y en DATO el valor 120. Si utilizamos la asignación:

TOTAL = TOTAL + DATO

Queda

En primer lugar se hace la suma de la derecha, 458 + 120, TOTAL= 578 obteniendo como resultado el 578. Con este proceso, los valores

almacenados solamente son usados en la operación.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

En segundo lugar, el resultado obtenido, 578 se almacena en TOTAL, con lo cual obligatoriamente se "pisa" el valor anterior (458) y se reemplaza por el nuevo.

Para observar este movimiento, podemos armar una tabla:

Estado		DATO	TOTAL	Memoria trabajo	de
anterior		120	458		
Operación derecha (suma	а а)	120	458	 578	
Asignación resultado	de	120	578		

Las tablas de prueba también se llaman prueba de pizarrón.

Esta tabla, normalmente se denomina "prueba de pizarrón" porque sirve al que escribió el proceso para seguirlo y ver como varían los valores para ajustar si alguno está cambiando mal, o perdiendo valores.

Las asignaciones se pueden clasificar de la siguiente forma:

CUIDADO... ¡!! Esto NO es cierto para las operaciones algebraicas en el campo de la matemática! En nuestro caso lo es, porque la asignación de la izquierda es POSTERIOR al cálculo de la derecha....

- **Simples**: Consiste en pasar un valor constante a una variable (a = 15)
- **De trabajo:** Es la que recibe el resultado de una operación que involucre muchas variables (datos) y operadores. Durante el proceso su valor cambia.
- Contador: Cuenta algún evento que se sucede, o mantiene datos de repeticiones
- **Acumulador:** Sirven para sumar los sucesivos valores que toma una variable.
- Banderas o marcas : Son variables que toman diferentes



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

valores dependiendo del estado del proceso, y luego son analizadas y utilizadas como condiciones para tomar decisiones.

Entonces

RESUMIENDO

- Las variables que aparecen a derecha del igual no se modifican al utilizarlas
- La que aparece a izquierda es UNA SOLA y su contenido anterior se borra, con lo cual decimos : *la asignación es destructiva*

Lectura:



La lectura consiste en recibir desde un dispositivo de entrada (p.ej. el teclado) un valor y almacenarlo en la memoria. Esta operación se representa en un pseudocódigo indicando el nombre del identificador del lugar de memoria donde se guardara el valor tecleado:

 ${\bf Ejemplo:}$

Leer dato

Leer a, b

Donde "a" y "b" son las variables

Obviamente, comparado con la *asignación*, siendo que la *lectura* "guarda" valores en un lugar de memoria, si había algo almacenado en ese lugar se borra para almacenar el valore leído. De allí que decimos que *la lectura es destructiva*

Escritura: Consiste en mostrar por un dispositivo de salida (p.ej. monitor o impresora) un resultado almacenado en un lugar de memoria (llamándolo por su identificador) o bien un mensaje. Este proceso se representa en un pseudocódigo como sigue:



Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Escribir "El resultado es:", R

Ejemplo:

Mostrar dato

Donde "El resultado es:" es un mensaje que se desea aparezca, por eso está entre comillas. R es un identificador de una variable que contiene un valor almacenado. Y la coma indica que se seguirá escribiendo al lado.

Con lo anterior se obtiene una línea con el mensaje contenido entre comillas, seguido por el valor almacenado en R.

La diferencia está en el contenido de las variables:

- -Para mostrar un **mensaje**, lo escribimos entre comillas
- -Para mostrar una **variable de caracteres**, se nombra simplemente en el lugar donde se quiere agregar a la cadena.
- -Para mostrar **un valor numérico** no puede agregarse tal como está almacenado, porque se deben indicar cuantos lugares ocupará, cuantos antes y cuantos después de la coma, o si deben cortarse o truncarse los decimales, así que veremos más adelante que se convierten a valores "alfanuméricos" y se indica la cantidad de espacio a disponer para mostrarlos.

Ante la pregunta del efecto que causa "mostrar" el valor de una variable, y de acuerdo a lo que hemos visto, se sobreentiende que el contenido es "leído" desde su lugar de memoria y enviado a la impresora o dispositivo, con lo cual **NO HAY EFECTO** sobre el valor almacenado (el contenido no se modifica)







Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS



UN EJEMPLO COMPLETO:

ATENCION!!

Analicemos un ejemplo completo. Vamos a calcular el área de un círculo. Para eso necesitamos saber :

o El radio del circulo

La *formula* a utilizar El radio del circulo será el dato que solicitaremos que se ingrese por pantalla, para que el algoritmo sea útil para todas las medidas de círculos. El calculo se hace sabiendo que $AREA = PI + RADIO^{2}$

o Así que, nuestro algoritmo (sencillo y lineal) sería: ⁷

Almacenar el valor Pi = 3.14

Definir en memoria DATO, AUX y AREA como valores reales(para almacenar los cálculos) Solicitar al usuario que ingrese el valor del radio por teclado

Guardar el radio en memoria en el lugar llamado DATO

Calcular el cuadrado de DATO y almacenarlo en

memoria en AUX

como calculo intermedio

Calcular el producto PI * AUX y almacenarlo en AREA (calculo final

Mostrar AREA en la pantalla

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

FIN

En un lenguaje más apto para nuestro uso, escribimos en pseudocódigo, que es una forma de escribir los procesos con palabras predefinidas, en forma simple, pero más adecuada para después pasarla a lenguaje de programación.

comenzar Pi = 3.14

DATO, AUX y AREA son reales

Leer DATO AUX = DATO * DATO

AREA = PI * AUX

Mostrar "EL AREA FINAL ES "; AREA

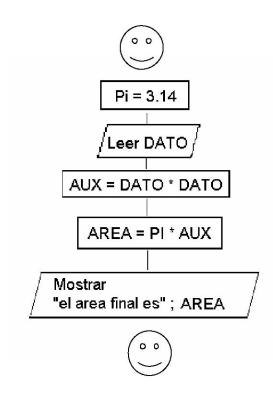
Fin

También podemos escribir nuestro proceso en un diagrama. Usaremos diagramas sencillos, donde los procesos que describimos se representan con rectángulos si son asignaciones y paralelogramos si son de entrada o salida.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

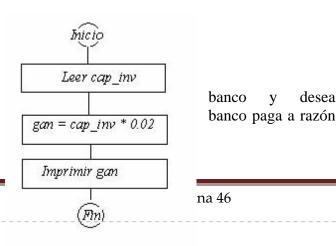


Aunque no es el símbolo correspondiente, hemos usado la carita para simbolizar el comienzo y el fin del proceso, tal como utilizamos en la nomenclatura de los diagramas representativos de los procesos de aprendizaje.

En todos los apuntes de informática, la letra corrida indica que estamos escribiendo en código para luego pasar a programar.

EJEMPLOS Y PROBLEMAS

Supongamos que un individuo tiene un capital en un saber cuanto dinero ganará después de un mes si el



desea

Introducción al Diseño de Algoritmos

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

de 2% mensual.

```
Inicio
Leer\ cap\_inv
gan = cap\_inv * 0.02
Imprimir\ gan
Fin
```

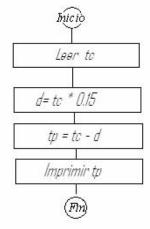
Hemos llamado **cap_inv** al capital que tiene la persona, y que se ingresará por teclado; **gan** es la ganancia mensual. Si quisiéramos calcular el monto final que retiraría la persona, tendríamos que sumar el capital y la ganancia, llamando gan_fin al lugar de almacenamiento:

```
Gan\_fin = cap\_inv + gan
```

Una tienda ofrece un descuento del 15% sobre el total de la compra y un cliente desea saber cuanto deberá pagar finalmente por su compra.

```
Inicio

Leer tc
d = tc * 0.15
tp = tc - d
Imprimir tp
```



Podrías decir con que nombre se han guardado los datos y los resultados ?

Un alumno desea saber cual será su calificación final en la materia de Algoritmos. Dicha calificación se

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

compone de los siguientes porcentajes:

55% del promedio de sus tres calificaciones parciales.

30% de la calificación del examen final.

15% de la calificación de un trabajo final.

Inici

Leer c1, c2, c3, ef, tf

$$prom = (c1 + c2 + c3)/3$$
 $ppar = prom * 0.55$
 $pef = ef * 0.30$
 $ptf = tf * 0.15$
 $cf = ppar + pef + ptf$

Imprimir cf

Fin

Aquí ha quedado sin hacer el diagrama... tendrás que hacerlo ...!

Problemas Propuestos

Esos problemas están para que los resuelvas solo. Si tienes dudas, pregunta a tu profesor.

- 1) Dada una cantidad en pesos, obtener la equivalencia en dólares, asumiendo que la unidad cambiaría es un dato desconocido.
- 2) Leer un número y escribir el valor absoluto del mismo.
- 3) La presión, el volumen y la temperatura de una masa de aire se relacionan por la formula:

$$masa = (presión * volumen)/(0.37 * (temperatura + 460))$$

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

FIN DEL CAPÍTULO 3: ACTIVIDADES DE FINAL DEL CAPÍTULO

Revisemos nuestros objetivos

Hemos tenido tanto trabajo!!

Que te parece si las preguntas de objetivos, las formulas vos? Acordate que siempre tienen que tener valores de respuesta SI ó NO y que tienen que servir para recordar los objetivos que están al principio del capítulo ... Adelante..! la tarea es tuya!

Nro Pregunta Verdad o no?

Recuperamos ideas:

- Revisá las definiciones de lenguaje y los tipos
- Mirá bien la definición de algoritmo
- ≥ Acordate que un buen algoritmo Siempre tiene que tener :
 - Inicio y final
 - Pasos simples
 - Solución única
 - Utilidad y eficiencia.

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Glosario del tema:

Programa

Lenguaje de programación

Programa fuente

Programa Objeto

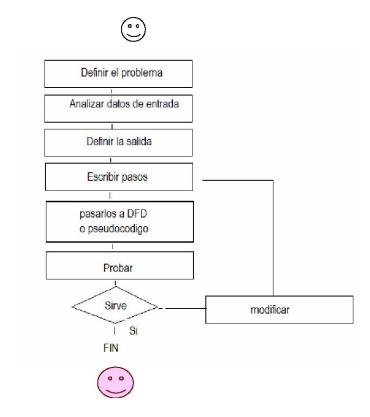
Algoritmo

Diagrama de Flujo

DFD



En el proceso de aprendizaje de la primera pagina... Resumimos

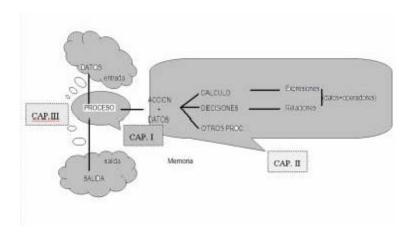


Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

Lo marcado en gris te indica hasta donde hemos avanzado!!



Hemos terminado ¡!

En este momento hemos terminado de aprender juntos un

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

capítulo importante en este curso. Para nosotros tu participación y actividad es muy importante. Es hora de que evalúes cuanto has entendido, y aprendido de todo esto. El próximo paso es hacer la evaluación prevista para este tema. No te quedes con dudas, relee los capítulos y comunícate con tu tutor...

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

BIBLIOGRAFIA

JEAN Paul Tremblay, B. Bunt Richard; "Introducción a la ciencias de las computadoras (enfoque algorítmico)" Mc Graw Hill

JOYANES Aguilar Luis; "Metodología de la programación" Mc Graw Hill

JOYANES Aguilar Luis; "Problemas de metodología de la programación" Mc Graw Hill

Materias Básicas

Introducción a la Informática -2008

DISEÑO DE ALGORITMOS

INDICE

Diagrama del contenido:	2
CAPITULO I. Repasando conceptos	3
Introducción	3
Lenguaje y comunicación	8
Y qué es un Algoritmo?	12
CAPITULO II. Elementos básicos	20
Los datos	21
Tipos De Datos	21
Operadores Aritméticos:	23
Constantes y Variables	31
Analicemos y agreguemos información útil	32
Fin del capítulo 2: Actividades de final del capítulo	34
CAPITULO III. Los procesos secuenciales	38
OBJETIVO:	38
Estructuras Secuenciales	39
Un ejemplo completo:	44
Ejemplos y Problemas	46
Fin del capítulo 3: Actividades de final del capítulo	49
INDICE	54