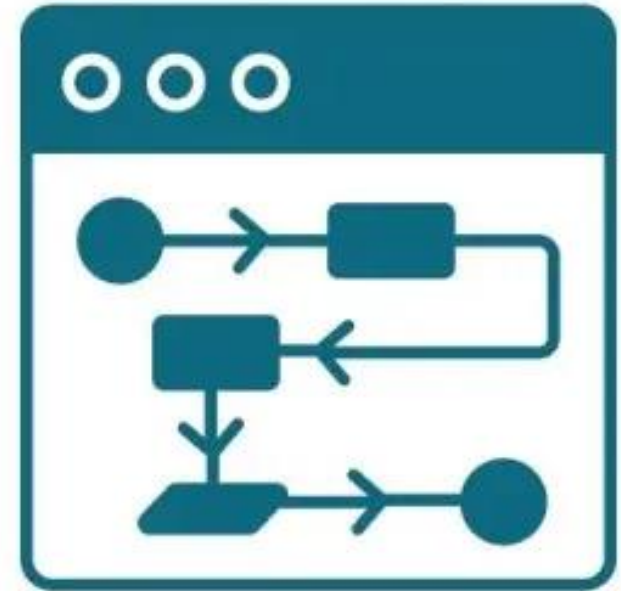




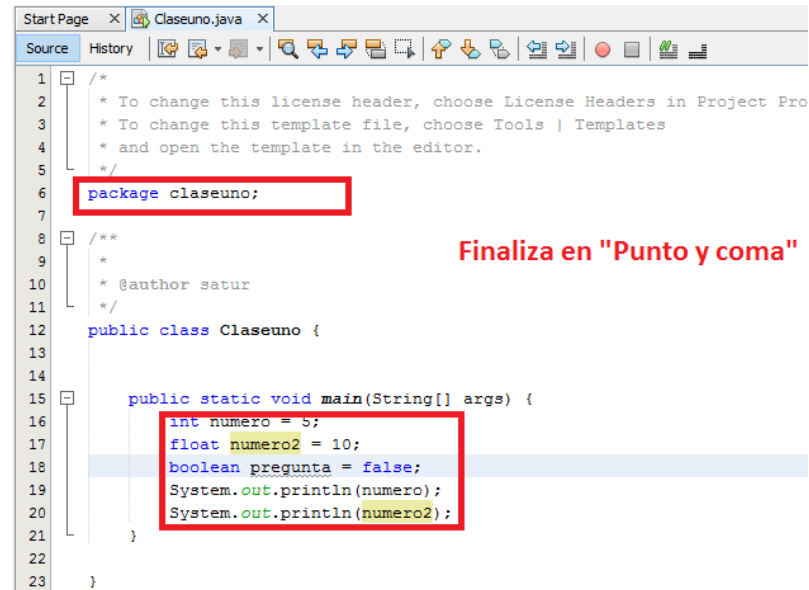
Algoritmos y Estructura de Datos



DOCENTE: MARX DANLY LEÓN TRUJILLO
FACULTAD DE ING. INDUSTRIAL Y SISTEMAS
E.P. INGENIERÍA DE SISTEMAS

ALGORITMOS Y ESTRUCTURA DE DATOS

- Operaciones con TDA
- Definición de Algoritmos
- Fases de resolución de problemas.
- Pseudocódigo



```
1  /**
2   * To change this license header, choose License Headers in Project Pro
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package claseuno;
7
8  /**
9   *
10   * @author satur
11   */
12  public class Claseuno {
13
14
15      public static void main(String[] args) {
16          int numero = 5;
17          float numero2 = 10;
18          boolean pregunta = false;
19          System.out.println(numero);
20          System.out.println(numero2);
21      }
22
23  }
```

Finaliza en "Punto y coma"

OPERACIONES CON ESTRUCTURA DE DATOS

RECORRIDO

Implica el acceder a cada registro una única vez aunque uno o más ítems del registro sean procesados. (Este acceso o procesamiento también se denomina a veces por el término «visitar» el registro).

BÚSQUEDA

Implica la localización de un registro caracterizado por una determinada clave o también el acceso a todos los registros que cumplan una o más condiciones.

INSERCIÓN

Cuando añadimos nuevos registros a la estructura.

OPERACIONES CON ESTRUCTURA DE DATOS

ELIMINACIÓN

Operación de borrado de un registro de la estructura.

ORDENACIÓN

Es la operación de clasificar los registros conforme a un orden lógico determinado (por ejemplo, alfabéticamente, de acuerdo a una clave de nombre, o numérica, de acuerdo a alguna clave de número, tal como número de Seguridad Social o de inventario).

MEZCLA

Es la operación de combinar dos archivos previamente ordenados en uno único que también lo está.

ALGORITMOS

Podemos decir que un algoritmo es un conjunto de operaciones (finitas y numeradas) cuyo objetivo es encontrar la solución de un problema o realizar un cálculo determinado.

Un ejemplo de algoritmo que seguimos todos los días (o deberíamos) es el proceso de lavarse los dientes:

- 1- Poner pasta dentífrica en el cepillo.
- 2- Humedecerla con agua.
- 3- Cepillar una muela hasta que esté limpia.
4. Cepillar otra muela..... hasta terminar el cepillado de todas las muelas y dientes.
5. Enjuagarse.
6. Limpiar el cepillo y secarlo.

ALGORITMOS

En el ámbito de la informática, los algoritmos son indispensables y son la esencia de todos los programas (software), de hecho un programa informático es un algoritmo (o un conjunto de ellos) traducido a un lenguaje de programación concreto (Java, C, Perl...).

La mayoría de las veces los algoritmos se escriben en pseudocódigo (es un híbrido entre el lenguaje natural y los lenguajes de programación), aunque también se pueden representar a través de diagramas de flujo o del lenguaje natural.

ALGORITMOS

EJEMPLOS

Ejemplo 1: Algoritmos de ordenación. Son aquellos cuya función es ordenar una lista de elementos según un orden determinado (alfabético, de menor a mayor, etc).

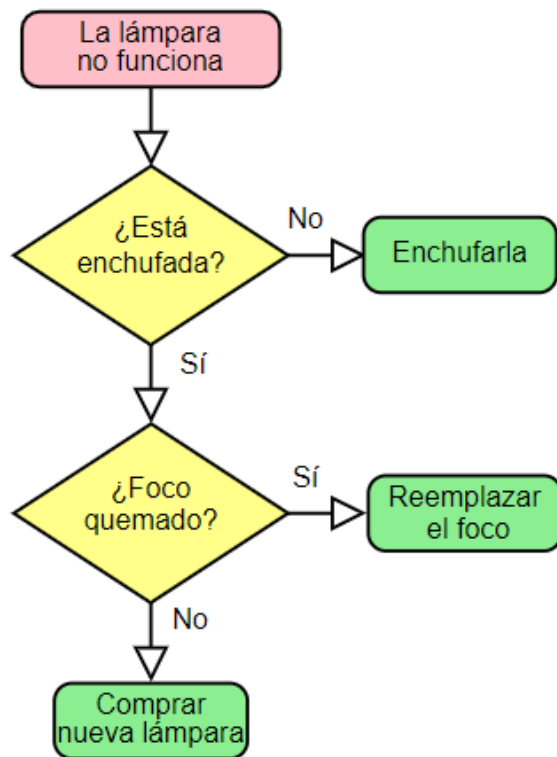
Ejemplo 2: De búsqueda: Dada una lista de elementos y un elemento a buscar, implementan la mejor manera de encontrarlo.

Ejemplo 3: Recursivos: Son los algoritmos que en algún momento se invocan a sí mismos.

Más ejemplos:

De la burbuja, backtracking, paralelos, búsqueda dicotómica, probabilísticos, determinísticos, metaheurísticos.

ALGORITMOS



algoritmo Sumar

variables

entero a, b, c

inicio

escribir("Introduzca el primer número (entero): ")

leer(a)

escribir("Introduzca el segundo número (entero): ")

leer(b)

$c \leftarrow a + b$

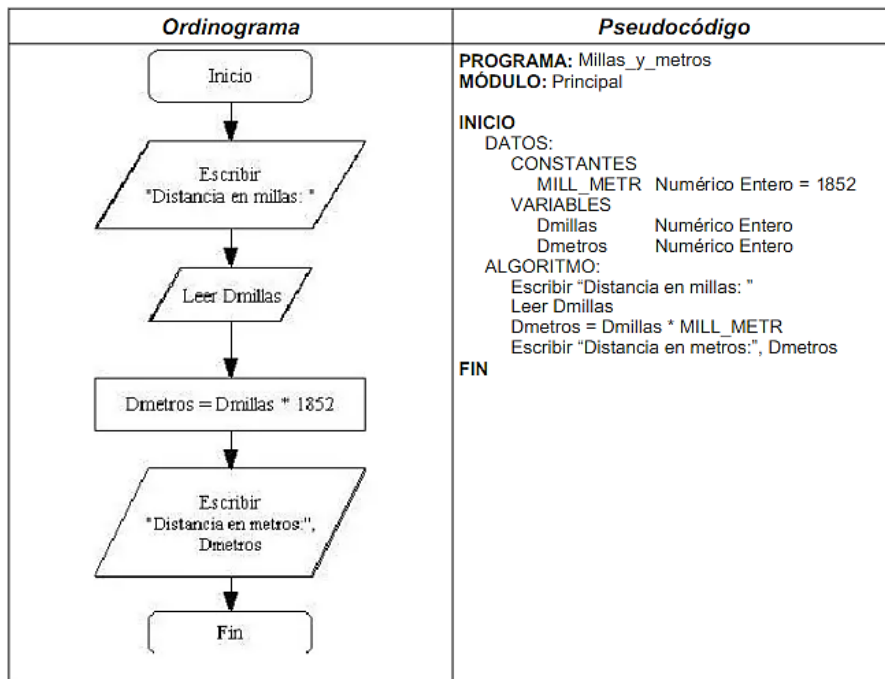
escribir("La suma es: ", c)

fin


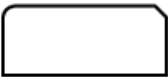
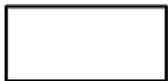


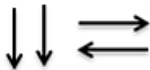
ALGORITMOS

Ejercicios resueltos

1. Diseñar el algoritmo (ordinograma y pseudocódigo) correspondiente a un programa que lea el valor correspondiente a una distancia en millas marinas y las escriba expresadas en metros. Sabiendo que 1 milla marina equivale a 1852 metros.



ALGORITMOS

SIMBOLOGIA	PSEUDOCODIGO	FUNCION
	Inicio o Fin	Se utiliza para empezar y terminar un programa
	Conocer o pedir	Utilizar para pedir datos por teclado estos pueden ser números, textos o datos alfanuméricos
	Evaluar o Asignación	Sirve para evaluar operaciones aritméticas como formular y también para asignar.
	Preguntas de decisión	Sirve para hacer decisiones.
	Imprimir	Imprime el Resultado
	Flechas de Dirección	Sirven para guardar la dirección del flujo del programa

FASES DE RESOLUCIÓN DE PROBLEMAS

El proceso de resolución de problemas con una Computadora conduce a la escritura de un programa y a su ejecución de la misma. Se puede considerar una serie de fases que generalmente deben seguir todos los programadores.



1. Análisis del problema



4. Compilación y ejecución



2. Diseño del algoritmo



5. Depuración y verificación



3. Codificación del programa



6. Documentación y mantenimiento

ANÁLISIS DEL PROBLEMA

- ❖ Es la **primera fase** en la resolución de problemas.
- ❖ Esta fase requiere una **clara definición**, donde se contemple exactamente **lo que se debe hacer** y el resultado o **solución** deseada.
- ❖ La **figura 5.1** muestra los **requisitos** que se deben definir en el análisis.

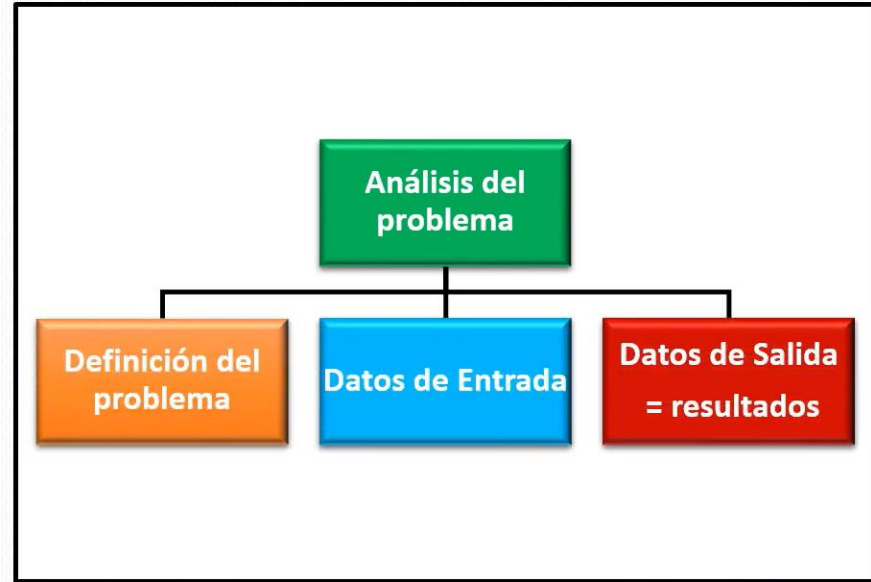


Figura 5.1: Análisis del Problema

DISEÑO DEL ALGORITMO

- ❖ En esta etapa se determina **cuáles son los métodos.**
- ❖ Un método conocido es el ***divide y vencerás***.
- ❖ Este método se conoce técnicamente como **diseño descendente** (*top-down*) o **modular**.

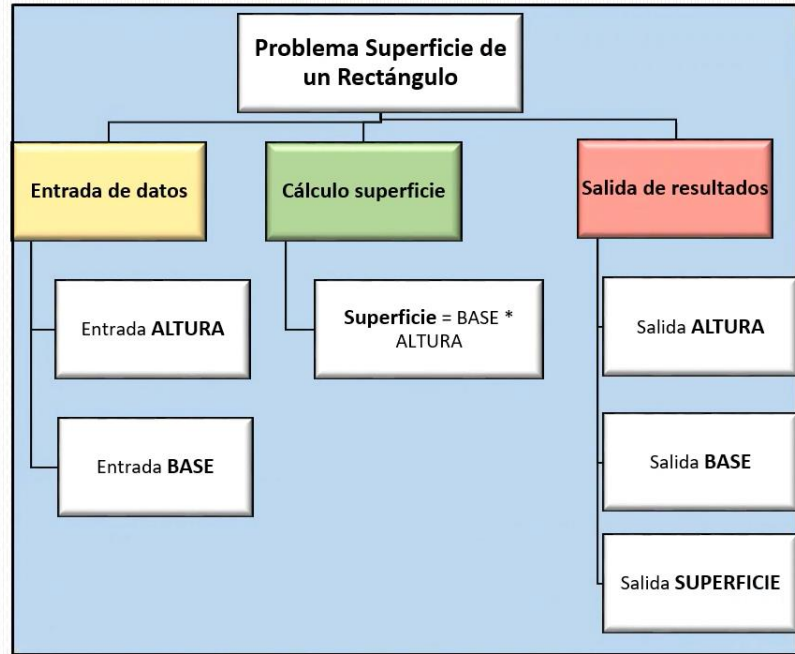


Figura 5.2: Diseño Descendente y Refinamiento Sucesivo

CODIFICACIÓN DEL PROGRAMA

- **Codificación**: es la escritura en un **lenguaje de programación** de la representación del algoritmo desarrollada en las etapas precedentes.

➤ Para realizar la **conversión** del algoritmo en programa se **sustituyen** las instrucciones indicadas en lenguaje natural por las mismas en el **lenguaje de programación** correspondiente.



```
Program Superficie; {Calculo de la superficie de un rectángulo}  
Use Crt;  
Var Altura, Base, Superficie: Real;  
Begin  
    ReadLn (Altura, Base);  
    Superficie := Altura * Base;  
    WriteLn (Altura, Base, Superficie);  
End.
```

Figura 5.3. **Codificación en Pascal**

COMPILACIÓN Y EJECUCIÓN

- ❑ El **programa fuente** debe ser traducido a lenguaje máquina, este proceso se realiza con el **compilador** o al menos que use un **intérprete**.
- ❑ Si tras la compilación se presentan **errores** (*errores de compilación*) en el *programa fuente*, es preciso volver a **editar el programa**, **corregir los errores** y **compilar** de nuevo.
- ❑ Cuando el **programa ejecutable** se ha creado, se puede ya **ejecutar** (*correr*) desde el sistema operativo con sólo teclear su nombre o hacer doble clic sobre el *programa ejecutable*. Suponiendo que no existen errores durante la ejecución (*llamados errores en tiempo de ejecución*), se obtendrán los resultados.



DEPURACIÓN Y VERIFICACIÓN

❖ La **depuración** es el proceso de **encontrar los errores lógicos** del programa y corregir o eliminar dichos errores.

❖ La **verificación** de un programa es el proceso de ejecución del programa con una amplia variedad de datos de entrada, llamados datos de prueba o test, que determinarán si el programa tiene errores («**bugs**»).



DOCUMENTACIÓN Y MANTENIMIENTO

❖ La **documentación del problema** consta de las **descripciones** de los pasos a dar en el proceso de resolución de un problema.

- La **documentación interna** es la contenida en líneas de comentarios.

- La **documentación externa** incluye *análisis, diagramas de flujo o pseudocódigos, manuales de usuario* con instrucciones para ejecutar el programa y para interpretar los resultados.



❖ El **mantenimiento del programa** consiste en mantener la **documentación actualizada** después de corregir errores o bien **modificar el programa**.

PSEUDOCÓDIGO

El pseudocódigo es una forma de escribir los pasos que va a realizar un programa de la forma más cercana al lenguaje de programación que vamos a utilizar posteriormente. Es como un falso lenguaje, pero en nuestro idioma, en el lenguaje humano y en español.

```
Proceso sin titulo
    totalfactura<-0;
    Escribir "Ingrese la fecha actual";
    Leer fecha;
    Escribir "Ingrese el nombre del cliente";
    Leer cliente;
    Escribir "Ingrese la direccion del cliente";
    Leer direccion;
    Escribir "Ingrese la cantidad de detalles";
    Leer detalles;
    Para i<-1 Hasta detalles Con Paso 1 Hacer
        Escribir "Ingrese la cantidad";
        Leer cantidad;
        Escribir "Ingrese la descripción";
        Leer descripcion;
        Escribir "Ingrese el precio";
        Leer precio;
        total<-cantidad*precio;
        totalfactura<-totalfactura+total;
    FinPara
    Escribir totalfactura;
FinProceso
```

PSEUDOCÓDIGO

Ejercicios de Pseudocódigo

1. Dados 10 números enteros que se ingresan por teclado, calcular cuántos de ellos son pares, cuánto suman ellos y el promedio de los impares.
2. Realizar un algoritmo en pseudocódigo para freír un huevo.
3. Hacer un pseudocódigo que imprima los números del 1 al 100 en orden decreciente.
4. Realizar un algoritmo en pseudocódigo que al introducir un número por teclado, nos diga si es positivo o negativo.
5. Realizar un algoritmo para definir el proceso de alistarse de una persona desde que se levanta.