

# OBJETIVOS DE LA ASIGANTURA: ALGORÍTMICA Y PROGRAMACIÓN I

- A. Potenciar la capacidad para la resolución de problemas y la expresión de soluciones mediante un lenguaje de programación estructurado, siguiendo la metodología top down, a través de :**
  - 1) Alcanzar alguna representación mental del mismo
  - 2) Convertirla en un modelo computable.
  - 3) Analizar alternativas de solución
  - 4) Decidir la secuencia de ejecución más eficaz y eficiente para cumplir con una función específica de un programa.
- B. Reconocer y resolver utilizando el lenguaje de programación, problemas de naturaleza Recursiva**

# CONTENIDOS MÍNIMOS

- Análisis y diseño de algoritmos.
- Tipos de datos y estructuras de datos elementales. Representación en memoria.
- Estructuras de control. Programación estructurada.
- Procedimientos y funciones.
- Recursividad.
- Algoritmos fundamentales: recorrido, búsqueda, ordenamiento, actualización.

# RESOLUCIÓN DE PROBLEMAS (usando una computadora)

- La resolución de problemas, utilizando como herramienta la computadora, requiere contar con la capacidad de expresión suficiente como para indicar a la máquina lo que debe llevarse a cabo.
- Hay un proceso de Abstracción entre la realidad y su representación en la computadora.

# Etapas en la resolución de un problema usando una computadora

- Análisis del problema
- Diseño de una solución
- Especificación de algoritmos
- Escritura de programas
- Prueba

# Análisis

Se analiza el problema en su contexto del mundo real. Deben obtenerse los requerimientos del usuario.

Del análisis surge un modelo del ambiente del problema y el objetivo a resolver.

Es la etapa del QUE queremos hacer y con que contamos.

## Diseño de la solución

Identificación de las partes ( subproblemas) en que el problema puede ser dividido.

Cada subproblema debe tener un objetivo específico ( es decir resuelve una parte del problema original)

La integración de todos los subproblemas debe permitir obtener la solución al problema.

# Especificación de algoritmos

La solución de cada subproblema debe ser especificada a través de un algoritmo.

Algoritmo: secuencia ordenada y finita de pasos elementales ( no ambiguos) que me llevan a resolver el problema.

La elección del algoritmo adecuado es fundamental para garantizar la eficiencia de la solución.

## Escritura de programas

Es la escritura del algoritmo en un lenguaje entendible por una computadora. ( Pascal, C, JAVA, ADA, etc.)



# Verificación

Verificar que su ejecución produce el resultado deseado.

# Algoritmo

Algoritmo: secuencia ordenada y finita de pasos elementales ( no ambiguos) que me llevan a resolver el problema.

Ejemplos:

P1: Indique la manera de salar una masa.

Malo: Agregarle un poco de sal.

Bueno: agregarle una cucharadita de sal.

Pre y Postcondiciones del algoritmo:

**Precondición.** Es la información que se conoce como verdadera antes de comenzar el algoritmo. ( En el ejemplo anterior, se cuenta con la masa, la cucharita y la sal)

**Postcondición:** es la información que se conoce como verdadera al concluir el algoritmo, si cumple adecuadamente con el requerimiento ( En el ejemplo anterior la masa debe estar salada).

# Estudios a realizar antes de la escritura de un programa:

- Estudio del Algoritmo que dará lugar al programa
- Estudio de los tipos de Datos y Estructuras de datos a utilizar

¿ Porque estudiar distintos tipos  
y estructuras de datos?

Para obtener una representación de  
la información que nos dé  
programas más Sencillos y  
Eficientes

# DATOS

Son una secuencia de símbolos, que representan algún atributo de una entidad.

( Clientes, Notas, Capacidad de un balde, etc)

# Abstracción de datos

Acción de separar las propiedades lógicas de los datos de sus implementación.

( permite tener una visión como programador distinta a la visión de la máquina, sólo secuencias de 0 y 1).

El proceso de Abstracción es posible porque los Lenguajes de computación Encapsulan los datos

# Encapsulamiento de Datos

Los lenguajes de programación y los sistemas operativos son los encargados de manejar esta conversión entre el mundo del programador y la realidad del Hardware de la computadora **Ocultando la información** de cómo se implementan los datos y sus tipos.

( Encapsulan los datos ocultando niveles de implementación)



# TIPO DE DATO

Está caracterizado por:

- Un rango de valores posibles
- Un conjunto de operaciones realizables sobre este tipo de dato
- Una representación interna.

( Ej. NUMEROS, ALFANUMÉRICOS, LÓGICOS)

# Definición de Tipos de datos

- Definidos por el lenguaje ( tipos predefinidos, ej. Real, Integer, Char, etc)
- Definidos por el Programador ( utilizamos recursos del lenguaje de programación para definir nuestro propio tipo, ej. String, Set, Subrangos.etc)

# Tipos de datos – Leng Pascal

- **ESTÁTICOS**

- **SIMPLES**

- **DEFINIDOS POR EL LENGUAJES**

- ORDINALES (Integer, Boolean, Char)

- NO ORDINALES ( Real)

- **DEFINIDOS POR EL PROGRAMADOR**

- (Enumerados, subrangos, Conjuntos y String)

- **ESTRUCTURADOS**

- ( Arreglos, Registros, Archivos)

- **TIPO PROCEDIMIENTO**

- **DINÁMICOS ( Punteros)**

# TIPOS ENTEROS PREDEFINIDOS

TIPO	RANGO
BYTE	0..255 ( 1 byte)
INTEGER	-32.768 .. 32.768 (2 bytes)
LONGINT	-247.483.648 .. 2.147.483.647 ( 4 bytes)
SHORTINT	-128 .. 127 (1 byte)
WORD	0 .. 65535 ( 2 bytes)

## OPERADORES MATEMÁTICOS:

**+, -, \*, DIV, MOD**

## OPERADORES DE RELACIÓN

**< ; <=; >; >=; <>**

# TIPOS REALES PREDEFINIDOS

TIPO	RANGO
REAL	$2.910^{-39} \dots 1.710^{38}$ ( 6 bytes)
SINGLE (REAL CORTO)	$1.510^{-45} \dots 3410^{38}$ ( 6 byte)
DOUBLE (REAL DOBLE PRECISION)	$5010^{-4932} \dots 1.710^{308}$ ( 6 byte)

## OPERADORES MATEMÁTICOS:

**$+$ ;  $-$ ;  $*$ ;  $/$**

## OPERADORES DE RELACIÓN

**$=$ ;  $<$ ;  $<=$ ;  $>$ ;  $>=$ ;  $<>$**

# OTROS TIPOS PREDEFINIDOS

- TIPO CARÁCTER (CHAR)

INCLUYE 1 DE LOS 256 CARACTERES DEL CODIGO ASCII  
(OPERADORES DE RELACIÓN =;<;<=;>;>=<>)

- TIPOS LÓGICOS (BOOLEAN)

INCLUYE LOS VALORES FALSE Y TRUE  
(OPERADORES LÓGICOS: AND;OR;NOT)  
(OPERADORES DE RELACIÓN =;<;<=;>;>=<>)

- TIPO CADENA STRING

ES UNA CADENA DE 0 A 255 CARACTERES.  
(OPERADORES DE RELACIÓN =;<;<=;>;>=<>)

# Ventajas de usar tipos de datos

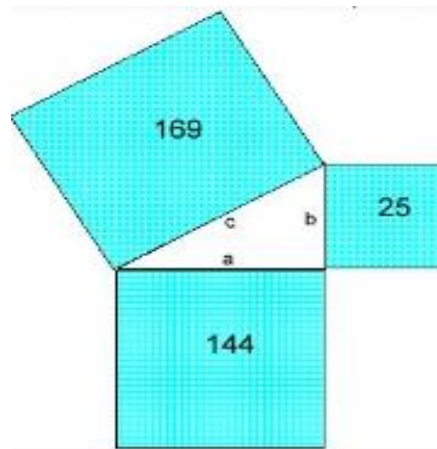
- **FLEXIBILIDAD** (modificando su definición, se modifican automáticamente las vbles. de ese tipo)
- **DOCUMENTACIÓN** (el nombre nos dice algo del tipo de dato)
- **SEGURIDAD** (permiten realizar testeo de código, usando o no directivas al compilador)

# EJERCICIOS:

RESOLVER LOS SIGUIENTES PROBLEMAS UTILIZANDO EL LENGAJE PASCAL:

Ejercicio 1:

Un lago que tiene la forma de un triángulo rectángulo, está circunscripto entre tres terrenos cuadrados de los que se conoce su superficie, tal como se presentan en la figura, se desea conocer la superficie del lago.





# EJERCICIOS

Ejercicio 2: Una agencia de autos paga a sus vendedores \$2500 de sueldo básico, más una comisión de 500 por cada auto vendido, mas un 5% del valor del vehículo. Que datos debo conocer para calcular el sueldo final