

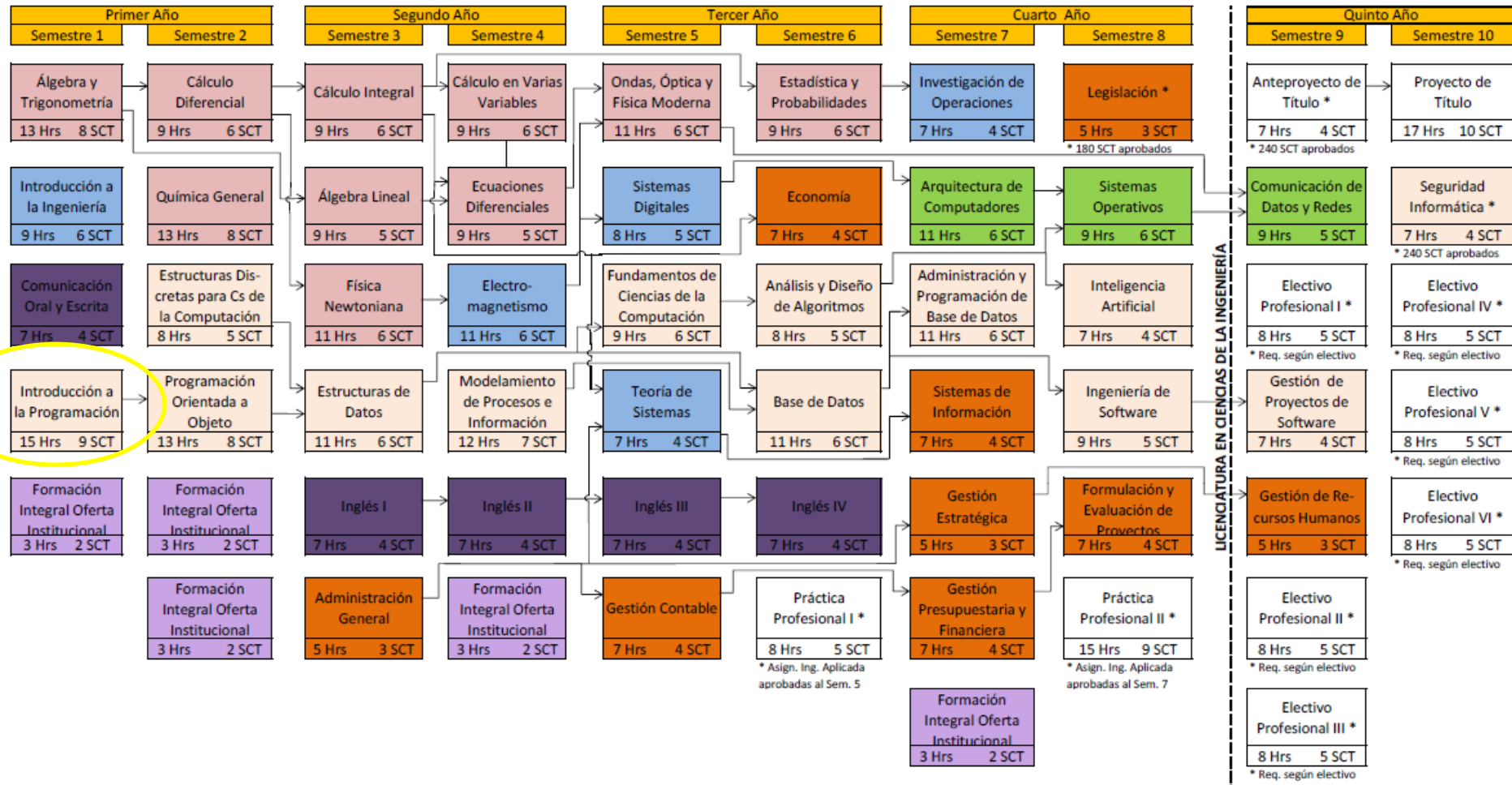
Ingeniería Civil en informática

Introducción a la Programación

Sección 1, 2 y 3
Profesor Horas Teóricas
Christian Vidal Castro
cvidal@ubiobio.cl

Marzo, 2023

MALLA CURRICULAR INGENIERÍA CIVIL EN INFORMÁTICA



Perfil del Ingeniero(a) Civil en Informática

El Ingeniero Civil en Informática de la Universidad del Bío-Bío es un profesional capacitado para comprender la problemática organizacional en el ámbito de la informática y, a partir de ello, plantear y desarrollar soluciones pertinentes. Lo anterior se traduce en la formulación, evaluación, e implementación de proyectos de tecnologías de información y comunicación.

Este profesional trabaja colaborativamente, lidera equipos multidisciplinarios en forma ética, proactiva, responsable y autónoma, respetando la diversidad de quienes conforman los equipos en que participa.

El sello distintivo del egresado de esta carrera lo constituye su conocimiento de la gestión organizacional y su disposición hacia el logro de metas, optimizando los recursos informáticos disponibles en la organización en la que se desenvuelva.

Perfil del Ingeniero(a) Civil en Informática

La formación profesional del egresado contempla las áreas de: Ciencias Básicas, Ciencias de la Ingeniería, Sistemas Computacionales, Ingeniería de Software y Base de Datos y Gestión de Informática.

Además, el egresado de esta carrera se distingue por el compromiso permanente con su aprendizaje, maneja el idioma inglés para apoyar el desarrollo de su trabajo y asume su quehacer profesional y ciudadano, sirviendo a la sociedad con innovación, excelencia y responsabilidad social.

Listado de Competencias

CE1: Gestionar sistemas computacionales para responder de forma óptima a los requerimientos de los usuarios evaluando su desempeño en base a los recursos disponibles.

CE2: Desarrollar software efectivo y eficiente, para diversos dominios, siguiendo un enfoque de ingeniería.

CE3: Construir bases de datos que satisfacen las necesidades de información de los usuarios, organizaciones o individuos, mediante el uso de diversas técnicas y herramientas.

CE4: Gestionar los recursos informáticos, de manera de apoyar y dar soporte a los procesos y estrategias de negocio de las organizaciones que permitan el mejoramiento continuo de las mismas.

CE5: Aplicar conocimientos de las ciencias básicas y de la ingeniería para resolver problemas usando pensamiento lógico racional y capacidades analíticas y de abstracción.

Relación de la Asignatura con las Competencias del Perfil de Egreso (1/4)

Esta asignatura permitirá al estudiante resolver problemas simples utilizando un lenguaje de programación. Se imparte en el primer semestre de la carrera.

Tributa a las subcompetencias específicas:

- **Analizar las problemáticas de las organizaciones** y de los individuos con el objeto de **determinar requerimientos de software** usando técnicas definidas para este propósito.
- **Resolver problemas de programación** utilizando lenguajes de programación y modelado de acuerdo a reglas y estándares existentes, y aplicando estrategias que aseguren la generación de soluciones eficientes.
- **Construir aplicaciones de software**, probando su funcionalidad y eficiencia, mediante el uso de arquitecturas, modelos, patrones, técnicas y herramientas de programación pertinentes para distintas plataformas.

Relación de la Asignatura con las Competencias del Perfil de Egreso (2/4)

Además, contribuye a la competencia genérica:

Manifestar una actitud permanente de búsqueda y actualización de sus aprendizajes, incorporando los cambios sociales, científicos y tecnológicos en el ejercicio y desarrollo de su profesión.

Relación de la Asignatura con las Competencias del Perfil de Egreso (3/4)

Descriptor de competencias

Resolver problemas simples de programación utilizando un lenguaje de programación procedimental.

Resultados de Aprendizaje:

- Explica los elementos sintácticos y semánticos de un lenguaje simbólico para posibilitar la construcción de algoritmos sintáctica y semánticamente correctos.
- Reconoce datos de entrada, salida, reglas de validación y acciones de transformación a partir del enunciado de un problema para la confección del algoritmo que permite resolver el problema.

Relación de la Asignatura con las Competencias del Perfil de Egreso (4/4)

Resultados de Aprendizaje:

- Construye algoritmos representados en un lenguaje simbólico para resolver problemas simples que tienen una solución algorítmica.
- Explica los elementos sintácticos y semánticos de un lenguaje de programación procedimental para posibilitar la construcción de programas sintáctica y semánticamente correctos.
- Construye programas en un lenguaje de programación procedimental para resolver problemas que tienen una solución algorítmica.

Programa de la Asignatura
Introducción a la Programación

Condiciones del Curso

Compromiso del profesor:

- Realizar todas las clases e iniciarlas puntualmente
- Informar oportunamente cambios en la planificación del curso
- Ver todo el programa y cumplir los objetivos del curso
- Intentar responder en forma clara todas las preguntas: en clase presencial, en foros, chat y correo electrónico
- Publicar todo el material en la plataforma educativa Moodle
- Publicar notas de controles a la semana siguiente de realizado el control, junto a su pauta de evaluación
- Definir y respetar horario de atención de alumnos

Condiciones del Curso

Compromisos del estudiante:

- Asistir a clases teóricas y Ayudantías
- Desarrollar los ejercicios planteados en las clases y módulos de laboratorios
- Prohibido copiar trabajos (de compañeros/as, Internet, etc.): En caso de detectar copias la nota será un 1 para todos los involucrados.
- Mantener siempre el respeto hacia los compañeros y los profesores
- En caso de inasistencia revisar el material de la clase y realizar trabajos solicitados
- Llegar puntualmente a las clases
- Entregar los trabajos en las fechas definidas

Evaluación

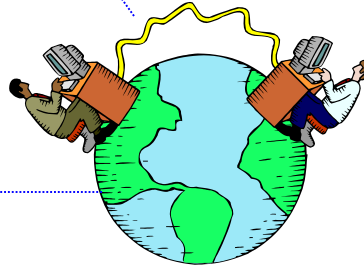
- 2 Test 30%
- 2 tareas 30%
- 1 Certamen 40%

¿ Por que usamos un computador?

En los niños...



En Comunicaciones..



En las compras...

En el trabajo...



En la medicina...

En el Cine..

En el Gobierno...

Etc.



En la Educación...



La informática está en todas partes

¿ Por qué se programa ?

Para resolver problemas !!

PROBLEMA: Situación del mundo real que puede ser resuelto/apoyado a través de un computador (Computabilidad).



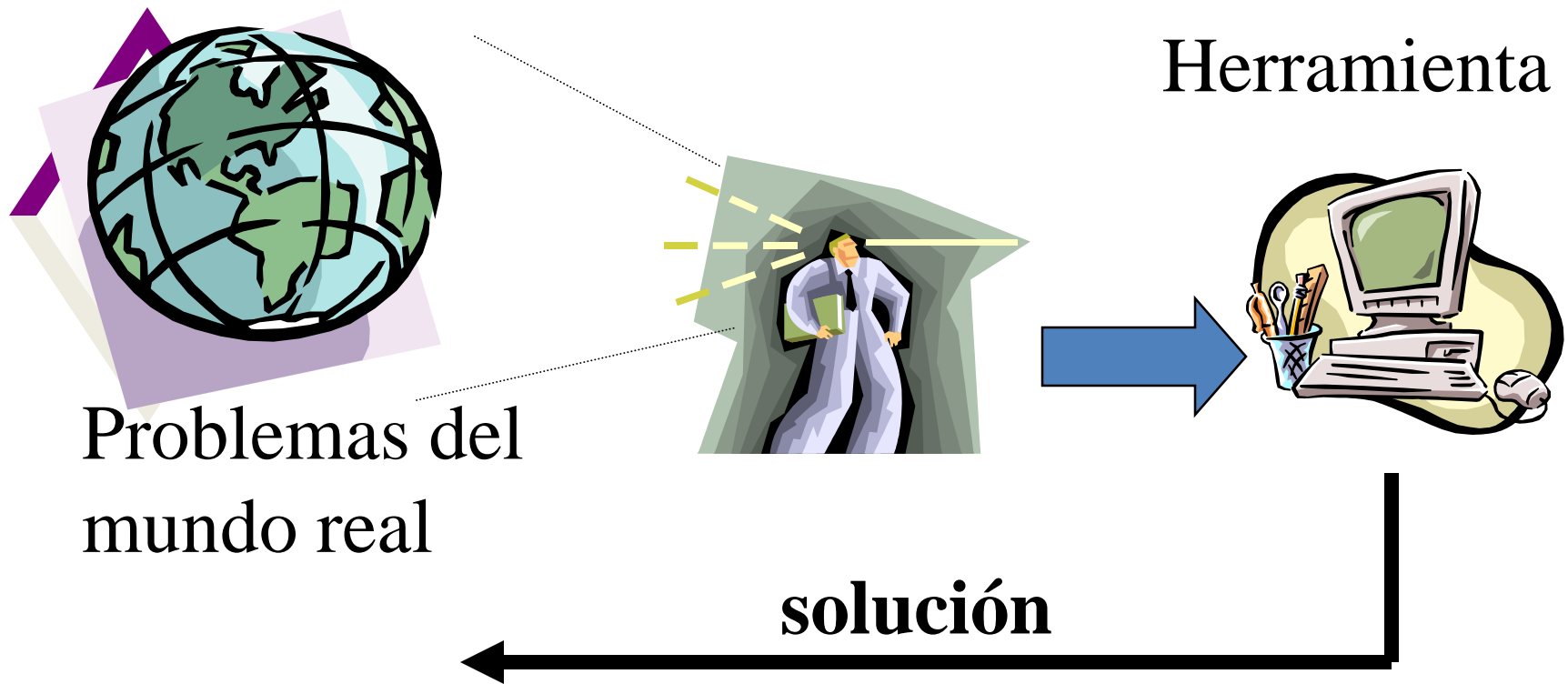
Para esta asignatura el “Problema” viene descrito mediante un Enunciado.

¿ Por que usamos un computador?

Herramienta de trabajo el cual está formado por dos componentes básicos: Hardware (circuitos) / Software (programas)

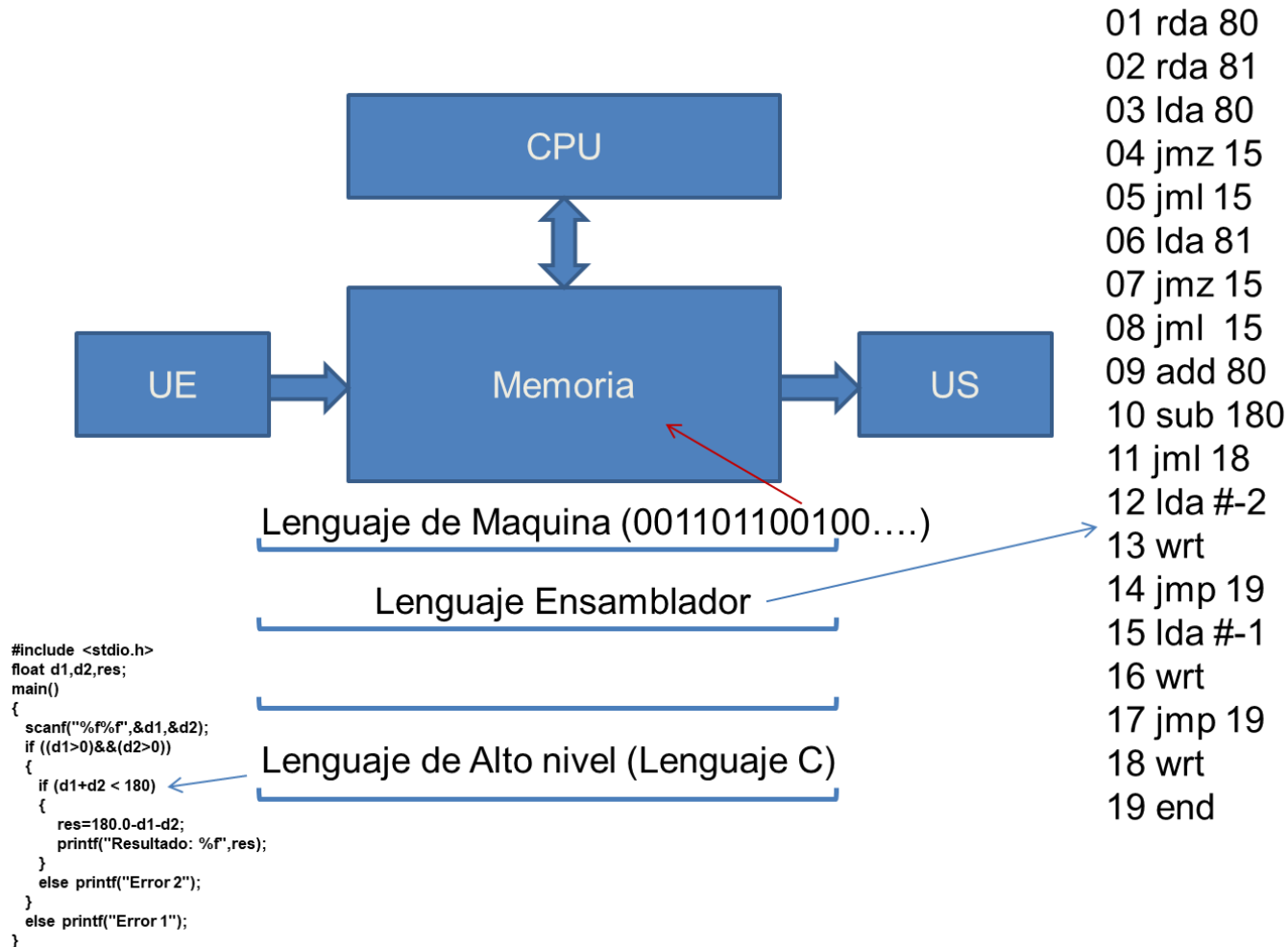


¿ Por qué usamos el computador?



¿Qué es un computador ?

Conjunto de instrucciones, con una secuencia lógica, escrito en algún Lenguaje de Programación que permite resolver un Problema. El programa recibe datos de entrada, realiza las operaciones de transformación requeridas, y entrega los resultados esperados.



¿ Qué es un lenguaje de Programación ?

Lenguaje artificial que puede ser usado para controlar el comportamiento de una máquina, especialmente un Computador. Estos se componen de un conjunto de reglas sintácticas y semánticas que permiten expresar instrucciones que luego serán ejecutadas por el procesador del computador.



INTRODUCCIÓN : Pasos para la CONSTRUCCIÓN de un programa

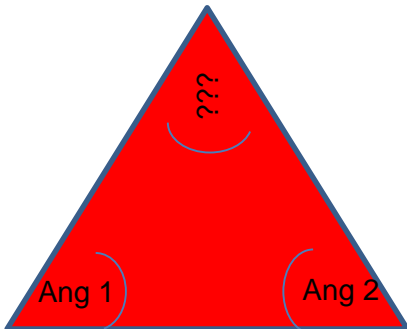
- Entender el problema (Contexto, resultados, transformaciones)
- Definir datos de entrada, salidas (información/resultados), reglas de validación, transformaciones
- Construir un Algoritmo (pseudocódigo, Diagrama de Flujo)
- ALGORITMO: Una secuencia de pasos (modelo) para realizar una tarea
- Probar el algoritmo
- Transformar el algoritmo en un programa escrito en un lenguaje de programación
- *Probar el programa (transformarlo a lenguaje de máquina y ejecutarlo)*

EJEMPLO

Problema/Enunciado: Construir un programa que permita calcular, dado dos ángulos de un triángulo, el valor del tercer ángulo.



- Entender el problema/enunciado (Contexto, resultados, transformaciones)
- Definir datos de entrada, salidas (información/resultados), reglas de validación, transformaciones



Transformación

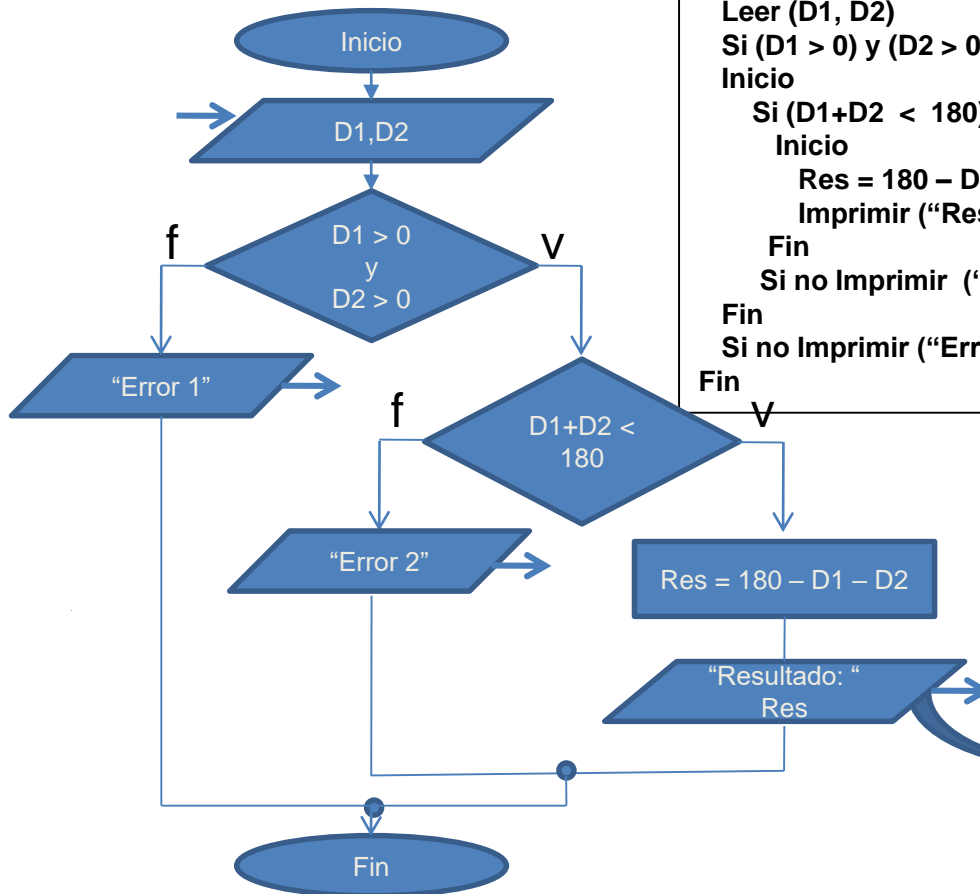
$$\text{Angulo 3} = 180 - \text{Angulo 1} - \text{Angulo 2}$$

Reglas de validación

- Ángulos ingresados > 0
- Suma de ángulos ingresados < 180

Ejemplo de Problema/Enunciado: Construir un programa que permita calcular, dado dos ángulos de un triángulo, el valor del tercer ángulo.

Algoritmo en Diagrama de Flujo



Algoritmo en Pseudocódigo:

Algoritmo Ejemplo

Variables

Reales D1, D2, Res

Inicio

Leer (D1, D2)

Si (D1 > 0) y (D2 > 0)

Inicio

Si (D1 + D2 < 180)

Inicio

Res = 180 - D1 - D2

Imprimir ("Resultado: ", Res)

Fin

Si no Imprimir ("Error_1")

Fin

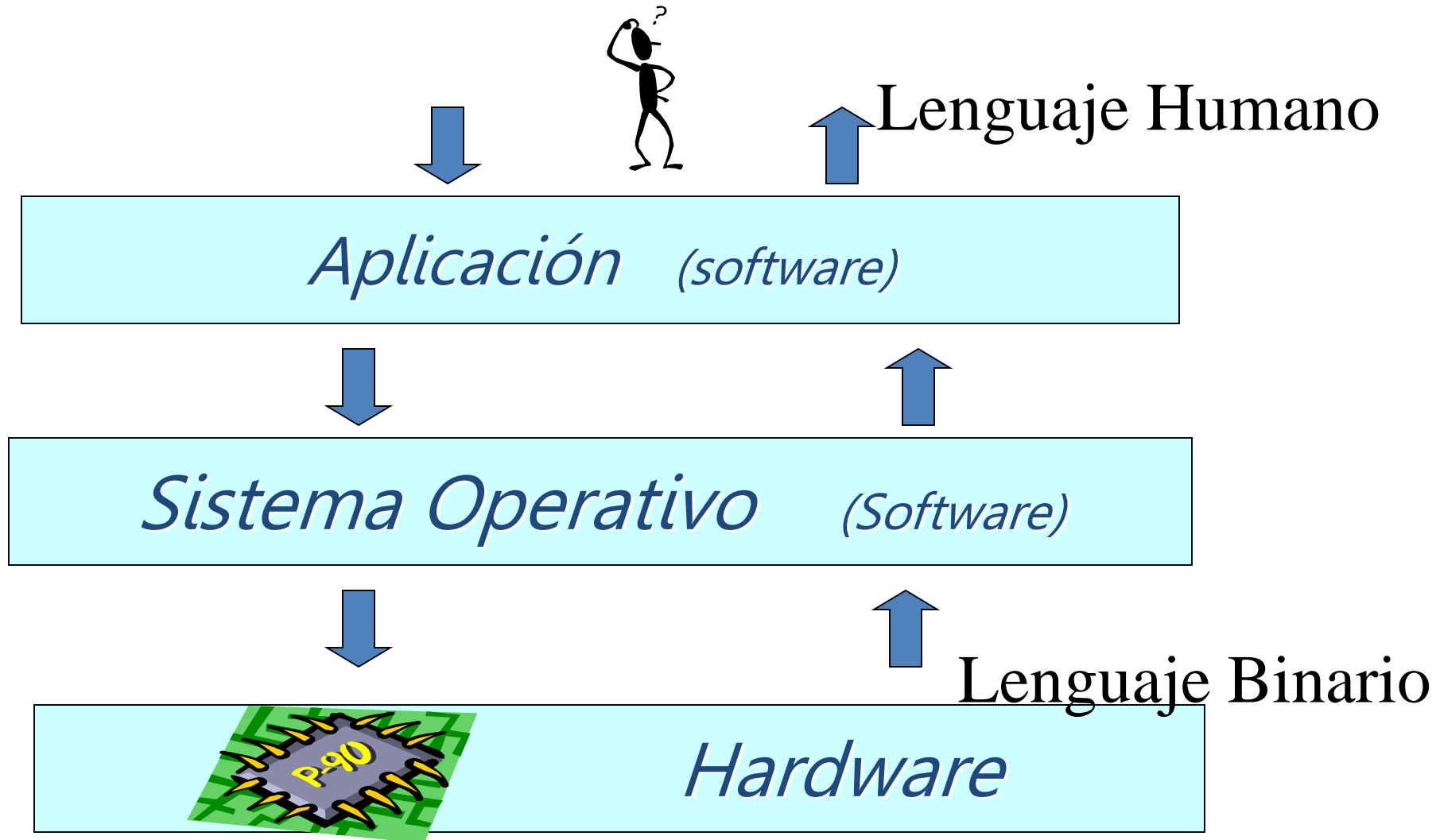
Si no Imprimir ("Error_2")

Fin

Programa escrito en
Lenguaje C

```
#include <stdio.h>
float d1,d2,res;
main()
{
    scanf("%f%f",&d1,&d2);
    if ((d1>0)&&(d2>0))
    {
        if (d1+d2 < 180)
        {
            res=180.0-d1-d2;
            printf("Resultado: %f",res);
        }
        else printf("Error 2");
    }
    else printf("Error 1");
}
```

¿ Cómo interactuamos con el computador?



¿ Qué es lo importante ?



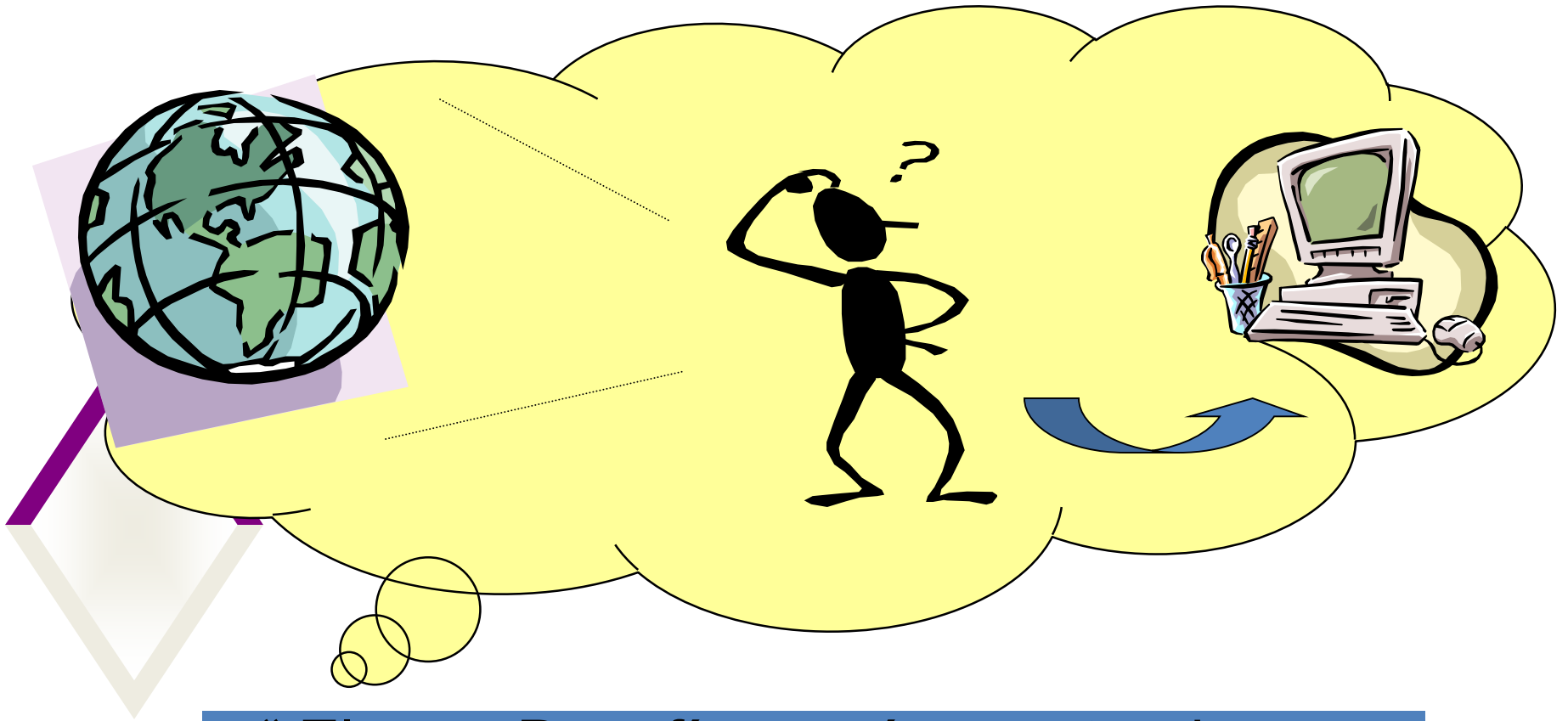
¿ Aprender cómo funciona el computador ?

¿ Aprender cómo usar el computador
para solucionar problemas ?

¿ Cual es el papel del computador ?



Convertir los datos de entrada en información mediante su procesamiento



“ El gran Desafío es cómo entender un problema, encontrarle una solución y cómo traspasar esa solución al computador para que la ejecute”

Entender un problema

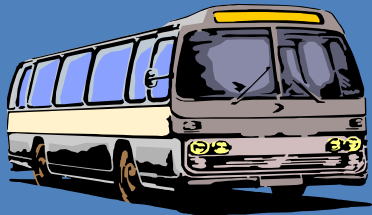
“Significar considerar sólo los aspectos de interés para el problema, el resto no interesa”.

(En realidad esta es casi la definición del concepto de abstracción muy usado en informática)



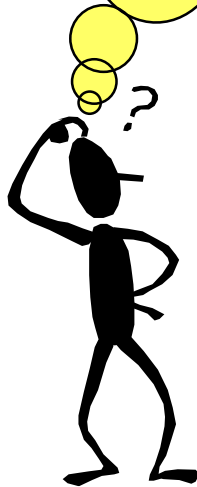
- No es una tarea fácil
- ¿Cómo se logra?
 - Análisis de las causas
 - Identificando datos (variables) involucradas y cuál es su comportamiento en el problema
 - Conociendo cuales son las partes o subproblemas que componen el problema total

Mundo Real



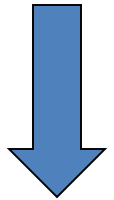
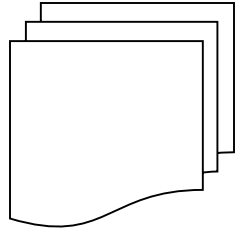
Problemas

IDEA
SOLUCIÓN



ALGORITMO

PROGRAMA



MEJORAS



Problema

PROGRAMADOR

*Análisis
del Problema*

*Diseño del
Algoritmo*

ALGORITMO

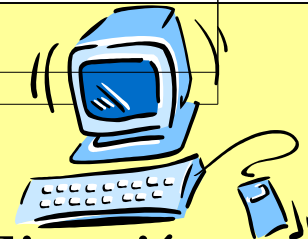
PROGRAMA

Codificación

**Sistema
de Computador**

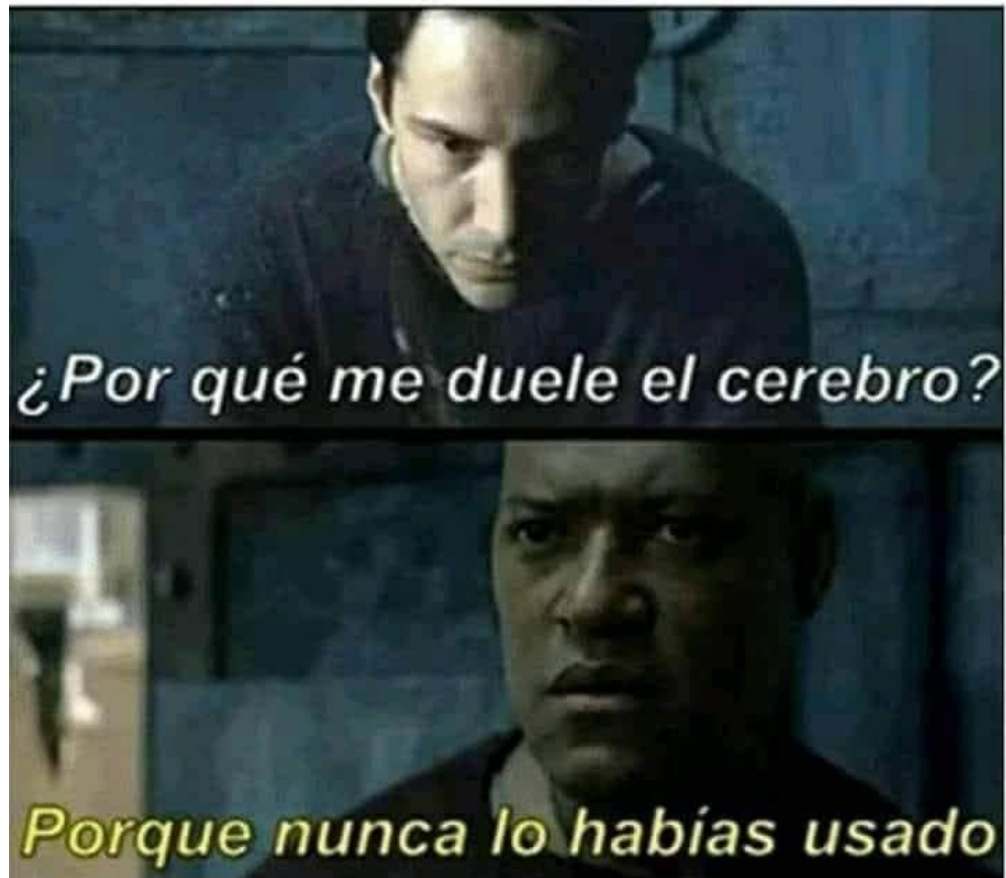
Traducción

**Ejecución
del programa**



Reflexiones previas sobre la programación

Cuando comienzas a estudiar programación



Reflexiones previas sobre la programación



Reflexiones previas sobre la programación

**CUANDO POR FIN TERMINAS DE
PROGRAMAR LA APLICACIÓN**



PERO ESTÁ LLENA DE BUGS

Reflexiones previas sobre la programación

"Ser programador no es estresante."



-Harold, 27 años.

Reflexiones previas sobre la programación

En este perfil amamos sufrir por eso
estudiamos programación 😊



Ejemplos: Comprensión de enunciado de problema

Determine datos de entrada, resultado esperado, reglas de validación y transformación para los siguientes enunciados.....

- Evaluar la función $f(x) = 3x^2 - 5x + 10$ para un valor de x
- Evaluar la función $f(x) = 3x^2 - 5x + 10$ para n valores de x
- Evaluar la función $f(x) = 3x^2 - 5x + 10$ para todo x entero en el intervalo $[a,b]$
- Imprimir la sumatoria de los n primeros números de la serie de Fibonacci

Consultas