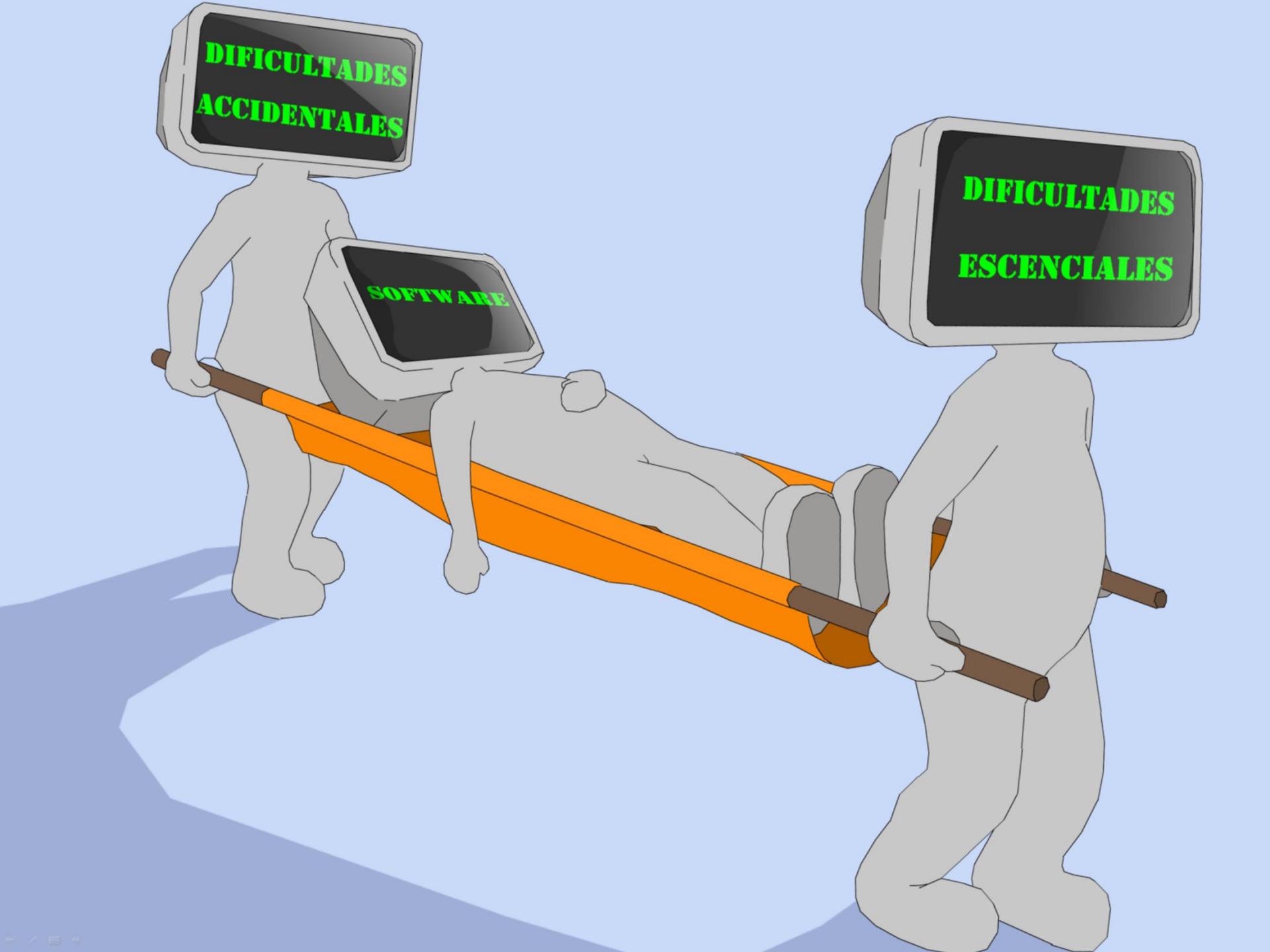


NO HAY BALAS DE **PLATA**





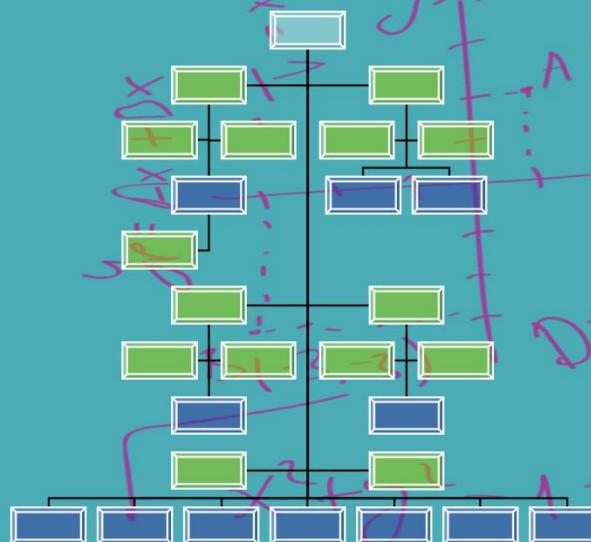
**DIFICULTADES
ACCIDENTALES**

SOFTWARE

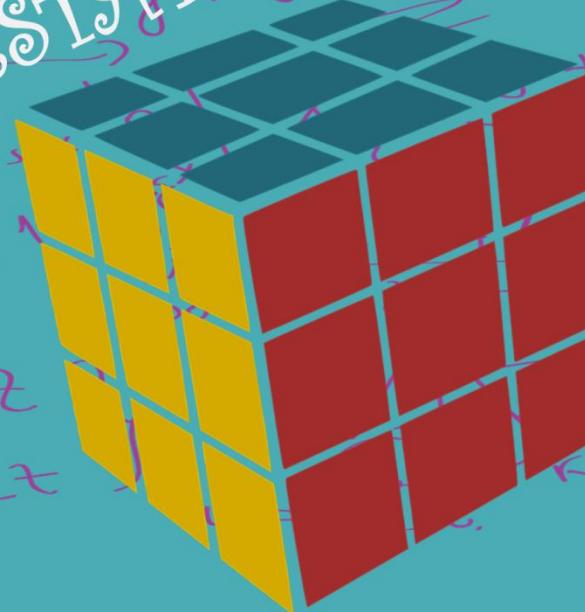
**DIFICULTADES
ESCENCIALES**

COMPLEJIDAD

Diagrama de Estados



ESTADOS



ADAPTA BILIDAD

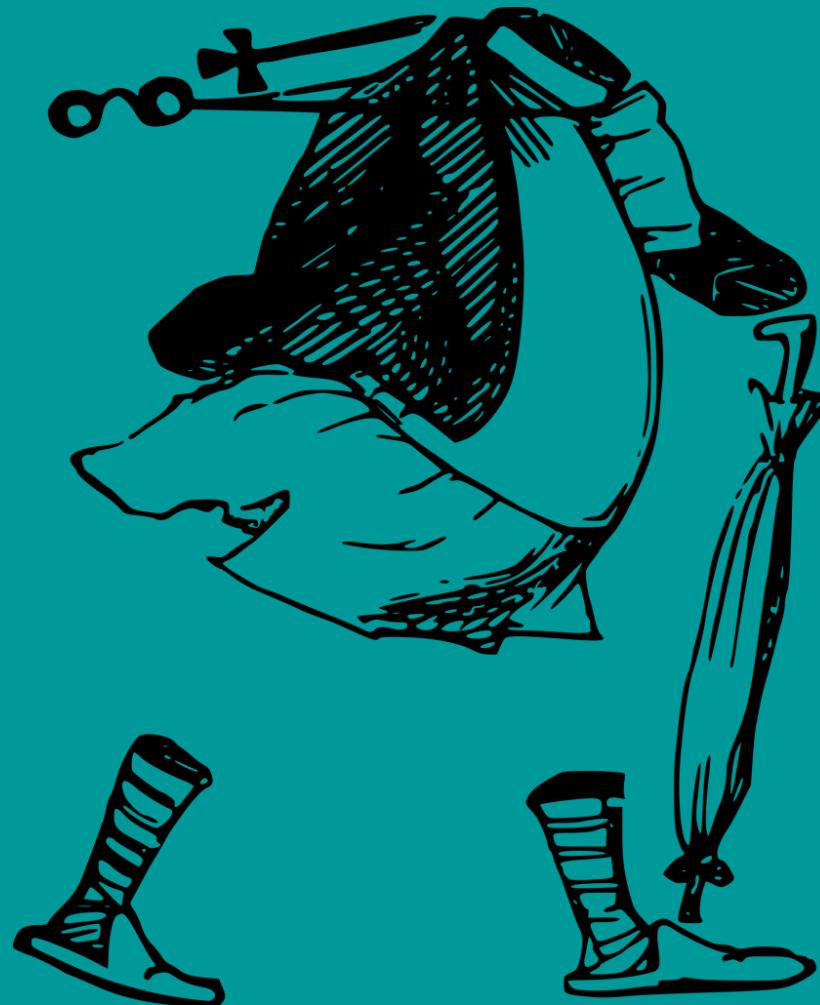


MODIFICABILIDAD



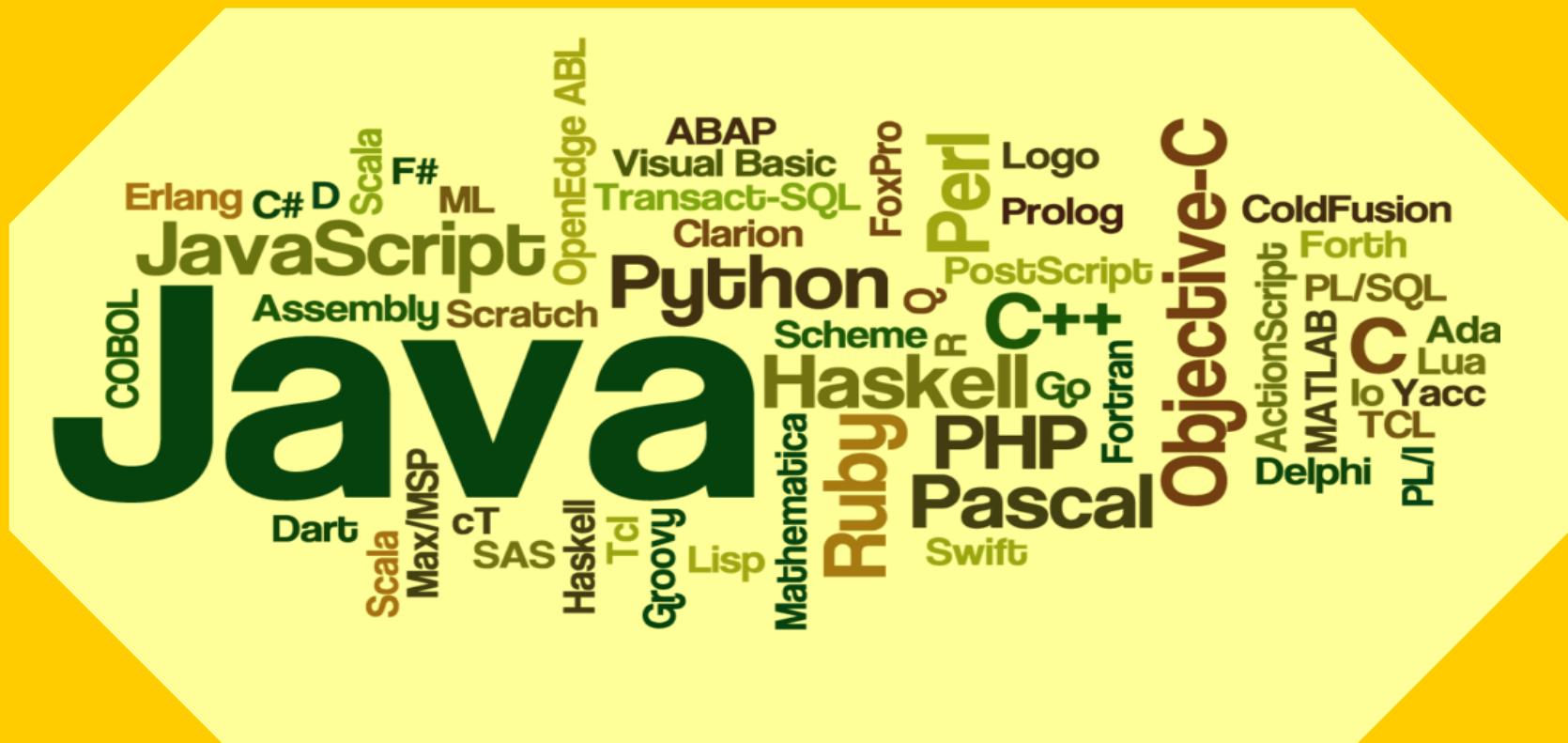
EL SOFTWARE DEBE ADAPTARSE AL HARDWARE

INVISIBILIDAD



AVANCES QUE RESOLVIERON DIFICULTADES ACCIDENTALES

LENGUAJES DE ALTO NIVEL



Tiempo Compartido



TIEMPO DE RESPUESTA

SISTEMAS DE INFORMACIÓN UNIFICADOS



It's
UNIX

Interlisp

Hopes for the Silver !



POO

J E R Á R Q U I C O S

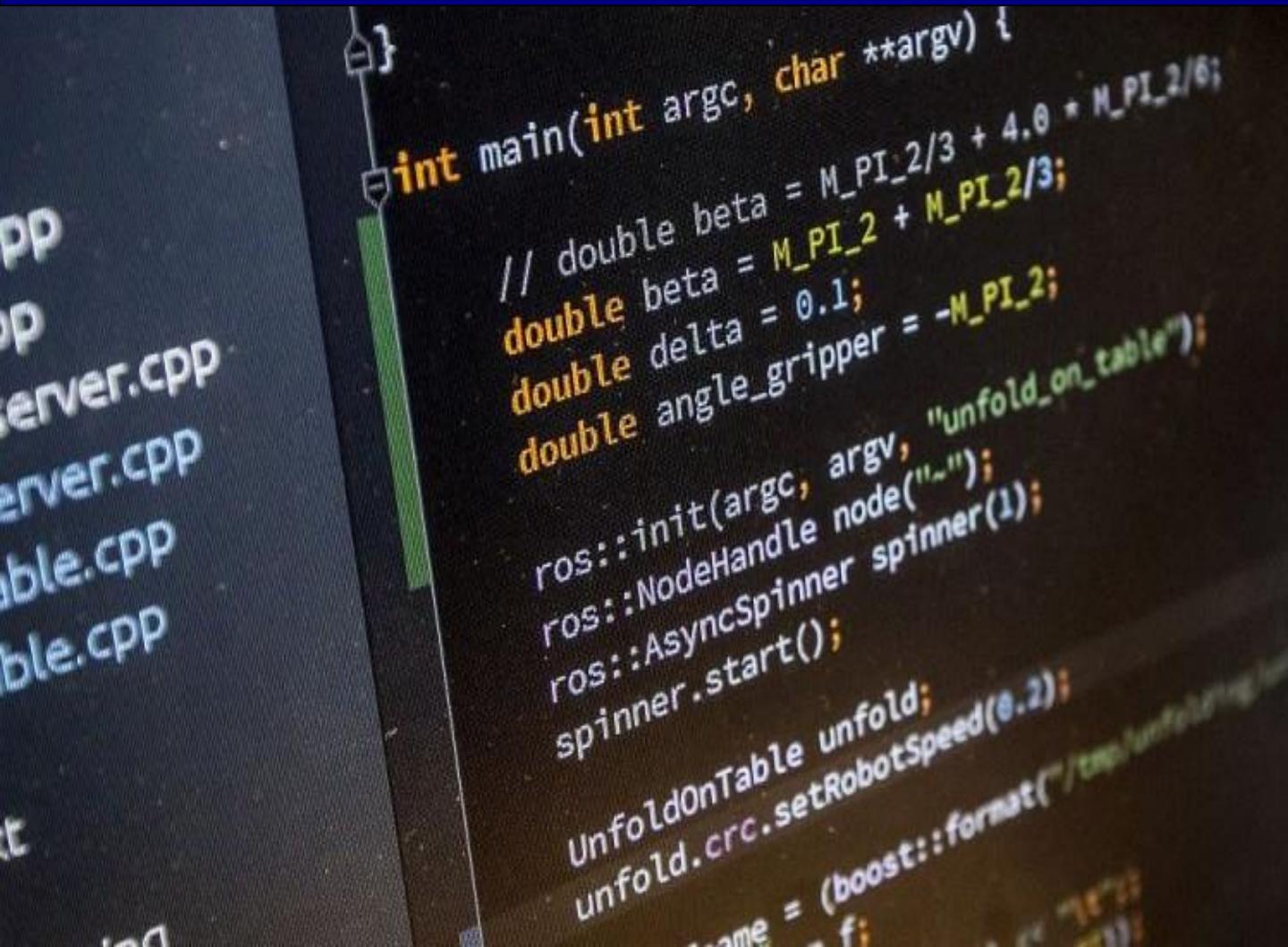
Tipos Abstractos

INTELIGENCIA ARTIFICIAL

..alcanza ?



PROGRAMACIÓN AUTOMÁTICA

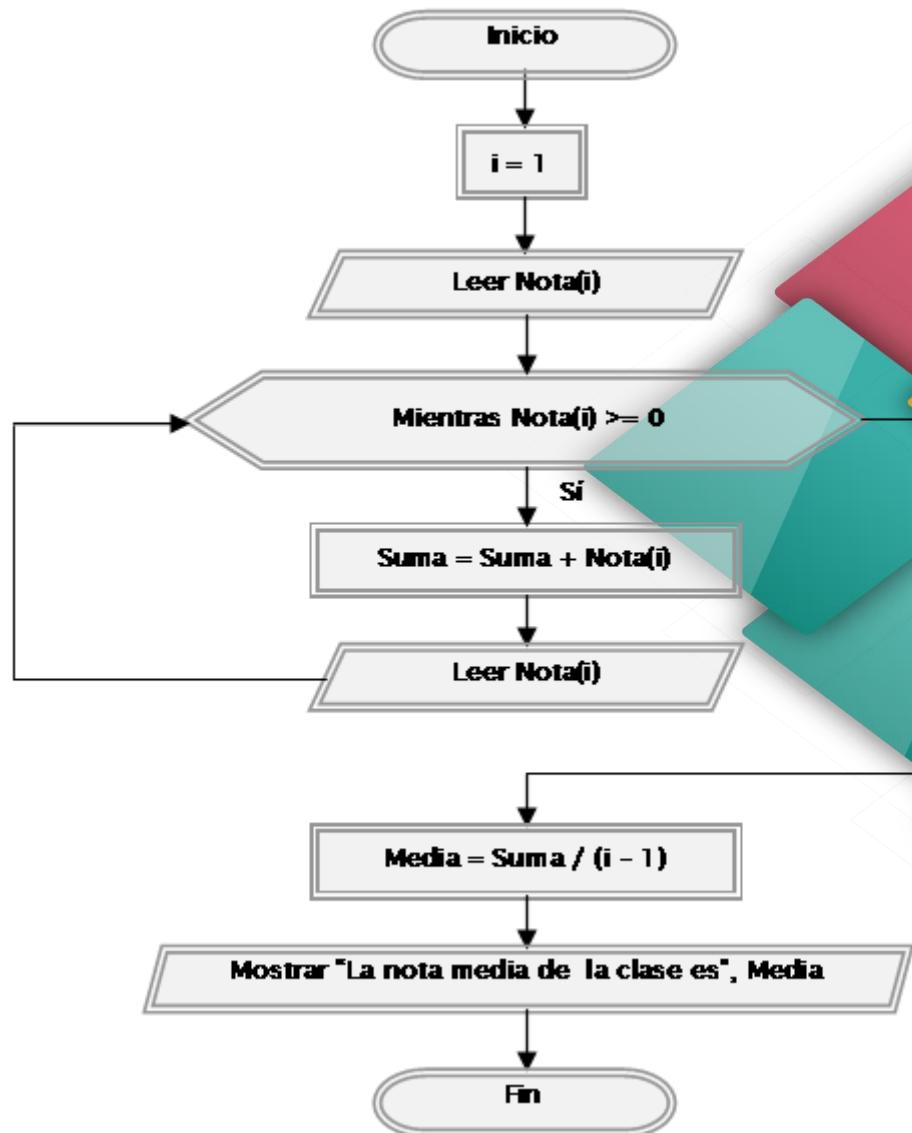


```
int main(int argc, char **argv) {
    // double beta = M_PI_2/3 + 4.0 * M_PI_2/6;
    double beta = M_PI_2 + M_PI_2/3;
    double delta = 0.1;
    double angle_gripper = -M_PI_2;
    ros::init(argc, argv, "unfold_on_table");
    ros::NodeHandle node("~");
    ros::AsyncSpinner spinner(1);
    spinner.start();
    UnfoldOnTable unfold;
    unfold.crc.setRobotSpeed(0.2);
    name = (boost::format("%s-%d") % name % f);
    name += ".f";
    if (f > 0) {
        if (f < 10) {
            name += "000";
        } else if (f < 100) {
            name += "00";
        } else if (f < 1000) {
            name += "0";
        }
    }
}
```

“Ha sido siempre un eufemismo para la programación en un lenguaje de más alto nivel que el disponible actualmente por el programador.”

Pamas

Programación



G

h

á

e

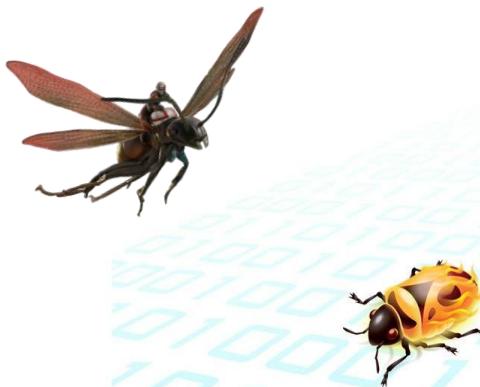
i

c

a

Verificación de programas

Dealing With Bugs



AMBIENTES Y HERRAMIENTAS



Ataques a las dificultades del software



*Dificultades
del Software*

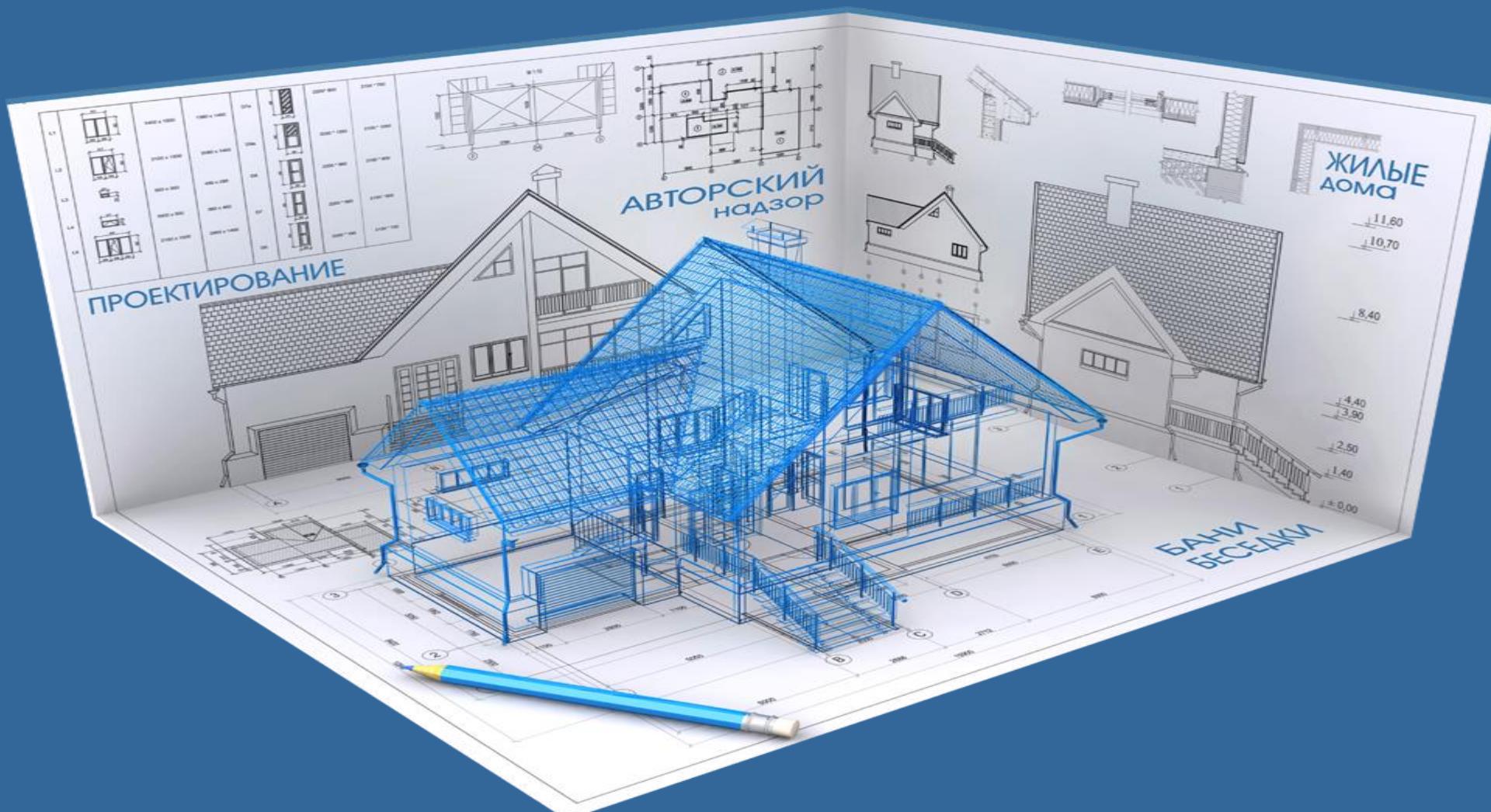
COMPRAR



CONSTRUIR

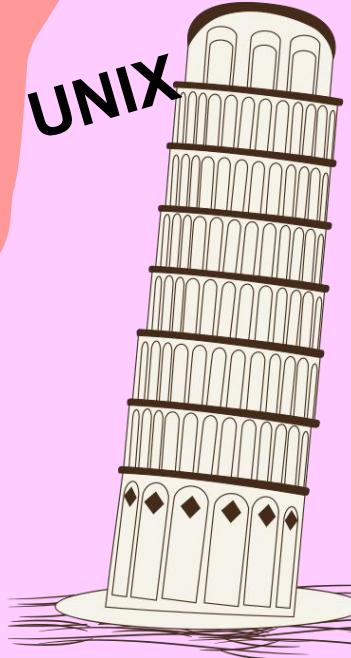


REFINAMIENTO DE REQUERIMIENTOS Y PROTOTIPOS DE INTERFAZ



GRANDES DISEÑADORES

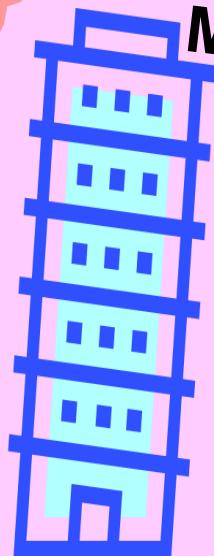
UNIX



PASCAL



MS-DOS



COBOL



No hay soluciones mágicas ...

