



# SIGNIFICADO DEL TRABAJO



# CONTEMPORANIDAD

- ↗ Modernidad y Pos-Modernidad
- ↗ La Libertad Estética
- ↗ Puritanismo y la Ciencia de la Administración
- ↗ Innovación y Creación



# DINÁMICA

```
import sh.net.*
import sh.media.*
import sh.utils.*
import sh.utils.Time

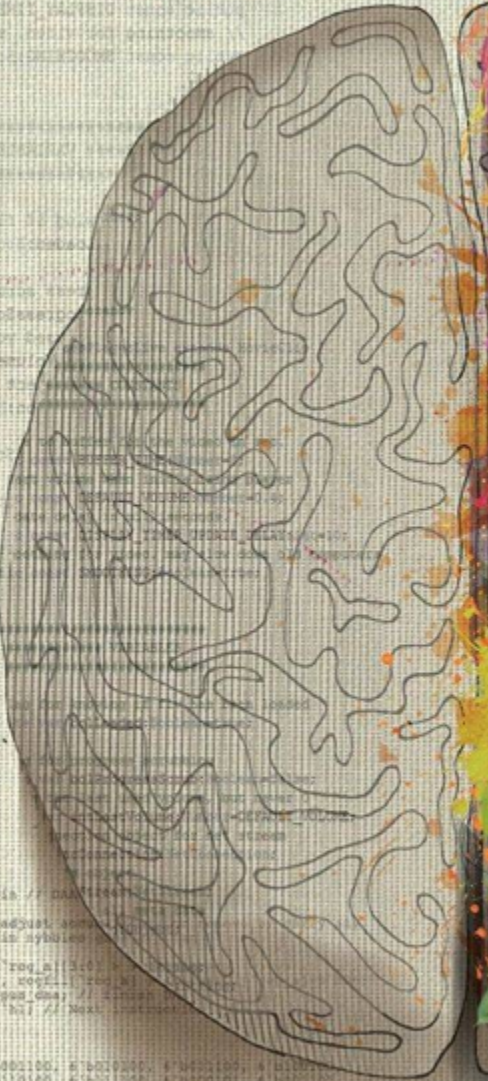
...

objInfo:Object;
strSource:String;
videoFileName:String;
bg_pic:MovieClip;

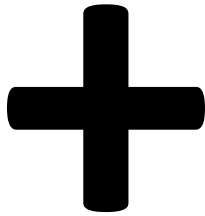
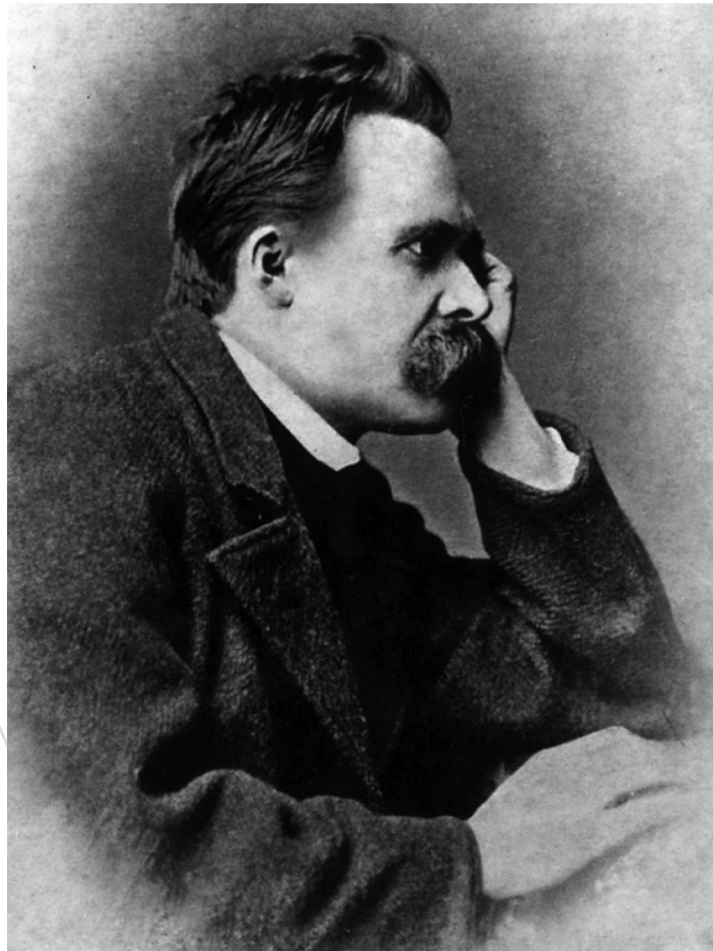
...

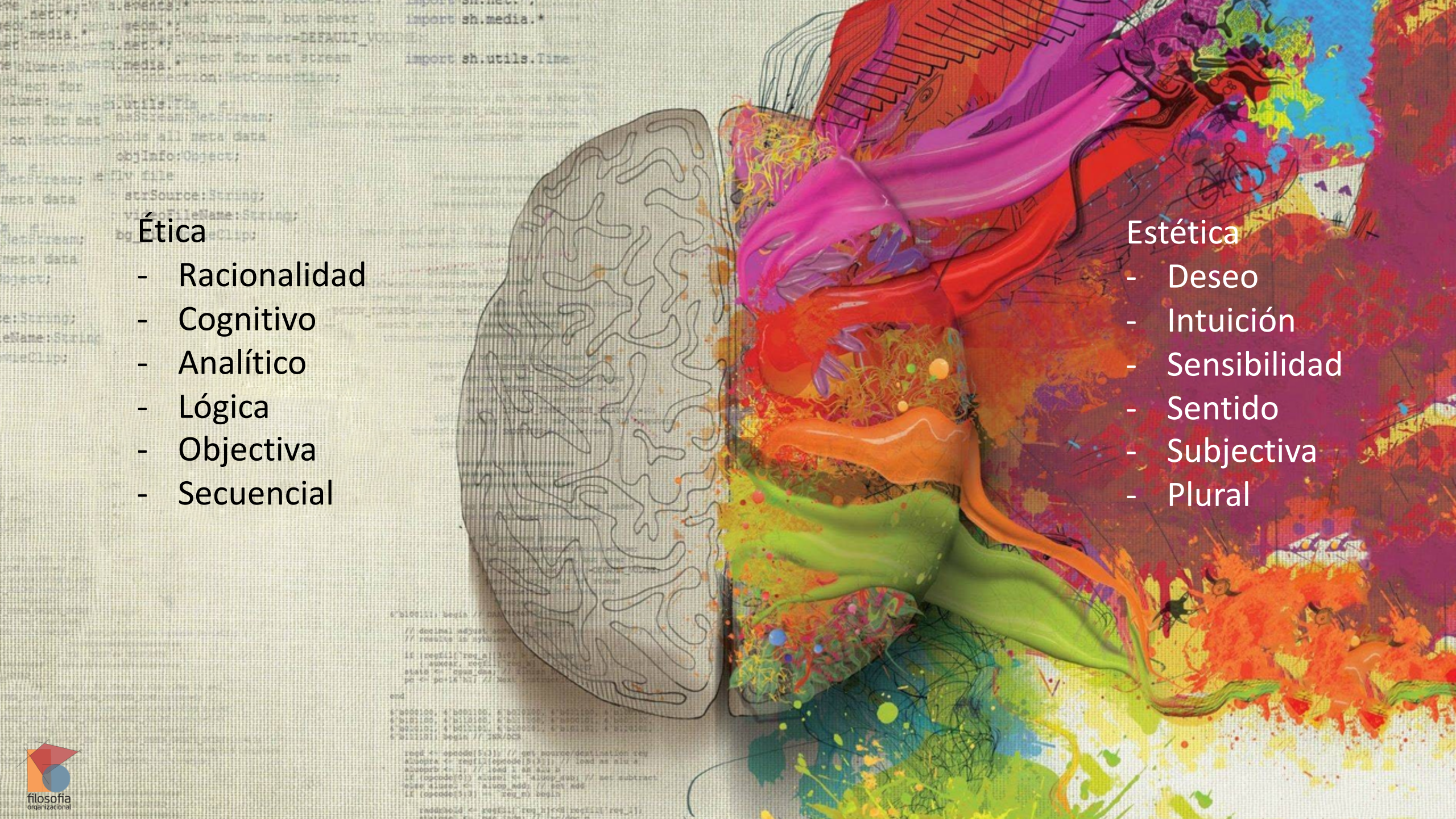
e:String;
eName:String;
eMovieClip;
```

```
0'b10011: begin // DAA/DAZ
// decimal adjust
// results in hybules
if (opcode[0] == 0)
  state <- opnd_csr; // 1st
  pc <- pc+16'h01 // Next
end
0'b000100: 0'b001000: 0'b010000: 0'b000100: 0'b100100: 0'b101000: 0'b110100: 0'b111000: 0'b000101: 0'b001001: 0'b010101: 0'b101101: 0'b110001: 0'b110101: 0'b111101: 0'b111101: begin // TBR/DBR
regd <- opcode[5:3] // get source/destination reg
alopndr <- regf[opcode[5:3]]; // load as alu &
alopndr <- 1; // load & alu &
if (opcode[0] == 0) aluop <- aluop_sub; // set subtract
else aluop <- aluop_add; // set add
if (opcode[5:3] == reg_r) obtain
raddrcsr <- regf[reg_r] << 16'h0001[reg_r];
alopndr <- raddrcsr;
```



# FILOSOFÍA Y NEUROCIENCIA





## Ética

- Racionalidad
- Cognitivo
- Analítico
- Lógica
- Objetiva
- Secuencial

## Estética

- Deseo
- Intuición
- Sensibilidad
- Sentido
- Subjectiva
- Plural

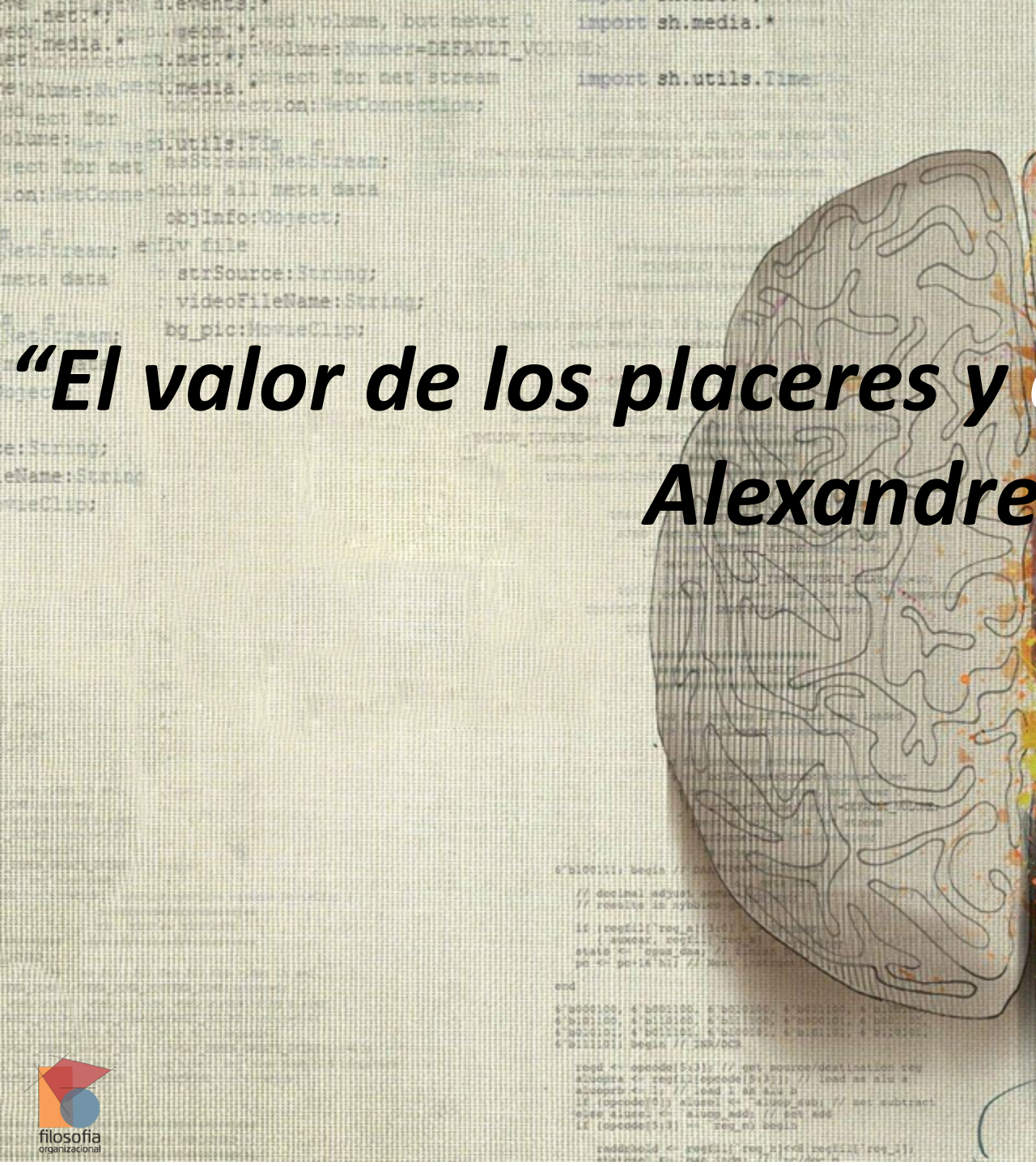
# EL SIGNIFICADO DEL TRABAJO

## Modernidad

- Poder del Capital Financiero
- Sacrificio
- Identidad
- Representación
- Alineación
- Eficacia
- Objetivación
- Poder
- Jefe Ejecutivo
- Admiración

## Contemporaneidad

- Poder del Capital Humano
- Placer
- Expresión
- Autenticidad
- Pluralismo
- Innovación / Creación
- Individualización
- Poder y Potencia
- Curador
- Admiración / Amor



***“El valor de los placeres y el placer de los valores”***  
***Alexandre Fialho***

```

import sh.media.*
import sh.utils.Time
import sh.Connection;
import sh.Utils;
import sh.Stream;
import sh.Connection;
import sh.Object;
import sh.File;
import sh.Data;
import sh.Video;
import sh.Pic;
import sh.String;
import sh.Name;
import sh.Clip;

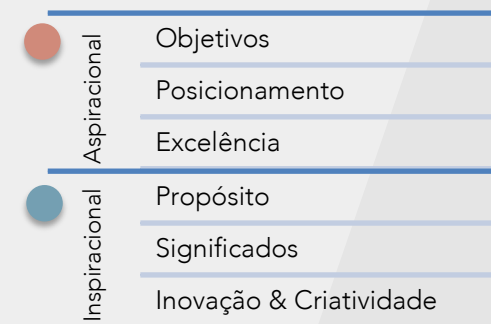
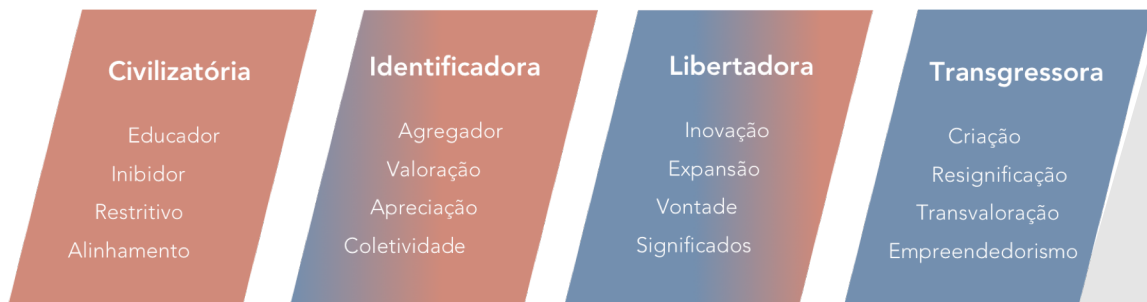
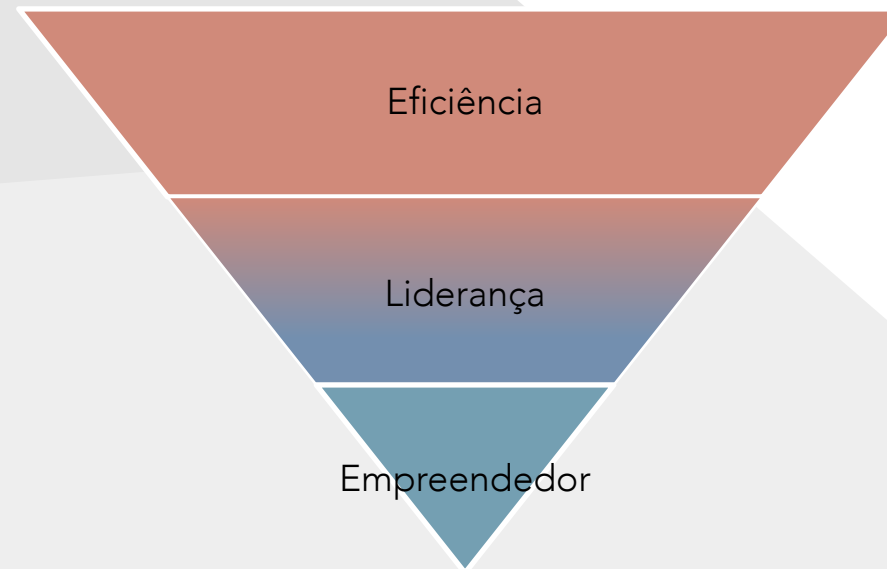
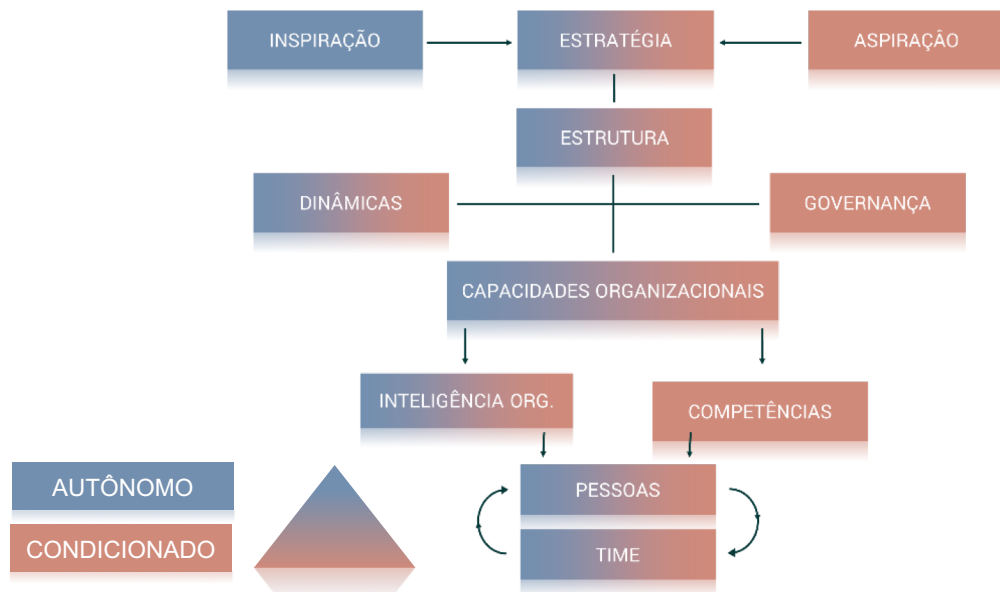
// decimal equal and 100
// results is always 100
if (reg[1] < reg[2])
    (auxvar, result) :=
        state <- opnd cba; // skip
        pc <- pc+16'0; // Next instruction
end
r000100, 4 b001100, 4 b010100, 4 b101100, 4 b110100
4 b100100, 4 b110100, 4 b110100, 4 b110100
4 b100100, 4 b110100, 4 b110100, 4 b110100
4 b111100, 4 b111100, 4 b111100, 4 b111100
4 b111100, 4 b111100, 4 b111100, 4 b111100
4 b111100, 4 b111100, 4 b111100, 4 b111100

reg <- opnd[5] // set source/destination reg
auxvar <- reg[1][opnd[5]] // load as aux &
auxopr <- 1 // load 1 as aux &
if (opnd[0])
    auxvar <- auxvar + auxopr // set subtract
else
    auxvar <- auxvar * auxopr // set add
if (opnd[5])
    reg <- auxvar

r000100 <- opnd[5] // set source/destination reg
auxvar <- reg[1][opnd[5]] // load as aux &
auxopr <- 1 // load 1 as aux &
if (opnd[0])
    auxvar <- auxvar + auxopr // set subtract
else
    auxvar <- auxvar * auxopr // set add
if (opnd[5])
    reg <- auxvar

```

# MODELOS CONJUGADOS







**GRACIAS**

Alexandre Fialho

fialho@filosofiaorganizcional.com.br