***Gobstones:***

Poner(color)

Mover(direccion)

IrAlBorde(direccion)

VaciarTablero()

nroBolitas(color)

opuesto(direccion)

opuesto(numero)

siguiente(direccion)

previo(direccion)

hayBolitas(color)

puedeMover(direccion)

Operaciones: +, -, \*, div, >, <, &&, ||, ==, <=, >=, not

function name(parameters)

{code return ()}

procedure name(parameters)

{code}

repeat(num)

{code}

if(boolean)

{code}

else

{code}

***JavaScript***

longitud(string)

convertirEnMayusucla(string)

comienzaCon(string, string)

imprimir(string)

tirarDado()

listasIguales(list, list)

longitud(list)

agregar(list, element)

remover(list, element)

posicion(list, element) da -1 si no está

function name(parameters)

{code}

if (boolean){code}

else{code}

let variable = value

for (let Elemento of Lista){}

Operaciones: +, -, \*, /, =, &&, ||, !, ===, !==, <, >, >=, <=, \*=, +=, -=, /=

Lista [], Registro{}, name.campo

***Ruby***

numero.abs

numero.times{code}

string.upcase

string.size

numero.even?

objeto.equal? elemento # = para asignación, == para objetos iguales, .equal? para si apuntan al mismo objeto

lista.push elemento

lista.delete elemento

lista.include? elemento

lista.size elemento

lista.select boolean {|elemento| condicion}

lista.find boolean {|elemento| condicion}

lista.all? boolean {|elemento| condicion}

lista.map code {|elemento| operacion}

lista.count boolean {|elemento| condicion}

lista.sum code {|elemento| value to add}

lista.each code {|elemento| code}

objeto = Clase.new #initialize number if needed

class name #< otraClaseParaHerencia

@variable

def name (parameters)

super #para mantener el código

end

end

class name

def initialize(name)

@name = name

end

def self.name (parameters)

end

end

if boolean

code

else

code

end

elsif boolean

end

lista = []

set = {}

procs: variable = proc {|parameters| code}

variable.call

getter devuelve el valor, usa el nombre del atributo

setter cambia el valor, usa el nombre del atributo con un =

para preguntar boolean termina en ?

para tomar acción termina en !

Operaciones: +, -, \*, /, &&, ||, ==, !=, <=, >=, <, >