

Ejercicio

El lobo feroz

El lobo feroz necesita un sistema para simular sus actividades diarias y poder saber si va a dejar de estar gordo (o “rellenito” como piensa él cada vez que se mira al espejo)

1) El lobo solitario

- Conocer la cantidad de calorías actual del lobo feroz.
- Que el lobo se pueda comer al chanchito (aporta tantas calorías como su peso dividido 10).
- Saber si el lobo está gordo (o sea, si tiene más de 200 calorías).
- Saber si el lobo está saludable (o sea, tiene entre 20 y 150 calorías)
- Hacer que el lobo corra un determinado tiempo (quema 2 calorías por minuto)

2) Caperucita Roja

- Representar la secuencia de mensajes que cuente la historia del Lobo y Caperucita, en una versión original:

El lobo va corriendo hasta el bosque, donde se encuentra con Caperucita y le pregunta a dónde va. Caperucita le dice que va a la casa de su abuela. Como tiene mucho hambre, el lobo se come a un personaje invitado que justo pasaba por allí.

Luego, el lobo corre a la casa de la abuelita y la come. Apenas ésta llega y le comienza a hacer preguntas molestas, el lobo también se come a Caperucita Roja.

El cazador nunca aparece... ¿Queda en un estado saludable el lobo? ¿Está gordo?

- Cuando el lobo come, sus calorías aumentan. Caperucita le aporta tantas calorías como su propio peso más el peso de su canasta, en la que hay una cierta cantidad de manzanas (cada manzana tiene 2 calorías). La abuela, está tan viejita que no le aporta ninguna caloría. El personaje invitado es alguien a elección (una oveja del cuento del pastorcito mentiroso, alguien que no les caiga bien, etc.), que tiene que poder ser comido por el lobo y debe aportarle una cierta cantidad de calorías, calculada de alguna forma creativa. Además, el lobo puede comerse a la abuelita y cualquier otro personaje a elección.
- El lobo, cuando va corriendo a un lugar consume calorías según lo que tarde en llegar. Se asume que el tiempo que demora en hacerlo depende únicamente del lugar donde vaya.

3) Los tres chanchitos

- Representar la secuencia de mensajes que cuente esta nueva versión de la historia de los 3 chanchitos:

El lobo sopla primero la casa de paja, la hace caer y el chanchito huye a la casa de madera donde estaba el otro chanchito. El lobo corre hasta la casa de madera, la sopla y la destruye, por lo que ambos chanchitos se van a la casa de ladrillos del tercer chanchito. El lobo los corre, sopla la casa de ladrillos y también logra tirarla

abajo. El lobo se come a los tres chanchitos. ¿Queda saludable el lobo? ¿Queda gordo?

- Cuando el lobo sopla una casa, pierde tantas calorías como la resistencia de la casa más el peso de los ocupantes. La casa de paja no resiste nada, la de madera tiene resistencia 5 y la de ladrillos resiste 2 por cada ladrillo.
- Inventar la forma en que se las diferentes casas se destruyen al ser sopladas por el lobo, procurando que sea diferente.
- ¿Qué otras cosas es necesario definir?

Solución Completa

<https://github.com/wollok/Polimorfismo-Lobo-Feroz>

Objetos

```
object lobo {  
  
    var calorías = 100  
  
    method calorías(){  
        return calorías  
    }  
  
    method estasGordo(){  
        return calorías > 200  
    }  
  
    method estasSaludable(){  
        return calorías > 20 and calorías < 150  
    }  
  
    method correr(minutos){  
        calorías = calorías - minutos * 2  
    }  
  
    method comer(personaje){  
        calorías = calorías + personaje.caloríasQueAporta()  
    }  
  
    method correrA(lugar){  
        self.correr(lugar.minutos())  
    }  
  
    method soplar(casa){  
        calorías = calorías - (casa.resistencia() + casa.pesoOcupantes())  
    }  
  
}  
  
object casaPaja{  
    method pesoOcupantes(){  
        return chanchito.peso()  
    }  
}
```

```

        method resistencia(){
            return 0
        }

        method minutos(){
            return 10
        }
    }

    object casaMadera{
        var minutos = 15
        var visitante

        method llega(alguien){
            visitante = alguien
        }

        method pesoOcupantes(){
            return chanchitoCasaMadera.peso() + visitante.peso()
        }

        method resistencia(){
            return 5
        }

        method minutos(){
            return minutos
        }
    }

    object casaLadrillos{
        var ladrillos = 10
        var pesoOcupado = chanchitoTrabajador.peso()

        method llega(chanchito){
            pesoOcupado = pesoOcupado + chanchito.peso()
        }

        method pesoOcupantes(){
            return pesoOcupado
        }

        method resistencia(){
            return 2 * ladrillos
        }
    }

```

```
        method minutos(){
            return casaPaja.minutos()+casaMadera.minutos()
        }
    }
```

```
object chanchito{
    method peso(){
        return 50
    }
    method caloríasQueAporta(){
        return self.peso() / 10
    }
}
```

```
object chanchitoCasaMadera{
    method peso(){
        return 20
    }
    method caloríasQueAporta(){
        return 7
    }
}
```

```
object chanchitoTrabajador{
    var edad = 2
    method peso(){
        return edad * 5
    }
    method caloríasQueAporta(){
        return self.peso()
    }
}
```

```
object bosque{
    method minutos(){
        return 10
    }
}
```

```
object canasta {
    var cantidadManzanas = 2
    method peso() {
        return 2 * cantidadManzanas
    }
}
```

```
object caperucita{

    method caloríasQueAporta(){
        return 50 + canasta.peso()
    }
    method dondeVas(){
        return casaAbuela
    }
}

object abuela{

    method caloríasQueAporta(){
        return 0
    }
}

object casaAbuela{
    var minutos = 5

    method minutos(){
        return minutos
    }
}

object oveja{
    var nombre = "Dolly"

    method caloríasQueAporta(){
        return nombre.size()*5
    }
}

object duende {
    var varitas = 0
    const altura = 5

    method encontrarVaritas(cantidad) {
        varitas = varitas + cantidad
    }

    method caloríasQueAporta() {
        return varitas + altura /10
    }
}
```

Tests

```
import loboFeroz.*

test "Las calorías del lobo son 80 cuando corre 10 minutos" {
    lobo.correr(10)
    assert.equals(80, lobo.calorias())
}

test "El lobo sopla casa de paja y sigue saludable" {
    lobo.soplar(casaPaja)
    assert.that(lobo.estasSaludable())
}

test "El lobo sopla casa de ladrillos y no queda saludable" {

    casaLadrillos.llega(chanchito)
    lobo.soplar(casaLadrillos)
    assert.notThat(lobo.estasSaludable())
}

test "El lobo sopla casa de madera, con los dos chanchitos dentro, y queda con 25 calorías"
{
    casaMadera.llega(chanchito)
    lobo.soplar(casaMadera)
    assert.equals(25, lobo.calorias())
}

test "El lobo sopla casa de ladrillos, con los tres chanchitos dentro, y queda con 0 calorías" {
    casaLadrillos.llega(chanchito)
    casaLadrillos.llega(chanchitoCasaMadera)
    lobo.soplar(casaLadrillos)
    assert.equals(0, lobo.calorias())
}
```

Program

```
import loboFeroz.*

program caperucitaYElLobo {

    console.println("Lobo al inicio: " + lobo.calorias())
    lobo.correrA(bosque)
    console.println("Lobo luego de correr al bosque: " + lobo.calorias())
    lobo.comer(oveja)
    console.println("Lobo luego de comerse la oveja: " + lobo.calorias())
    lobo.correrA(casaAbuela)
    console.println("Lobo luego de correr a la casa de la abuela: " + lobo.calorias())
    lobo.comer(abuela)
    console.println("Lobo luego de comerse a la abuela: " + lobo.calorias())
    lobo.comer(caperucita)
    console.println("Lobo luego de comerse a caperucita: " + lobo.calorias())
    console.println("Esta gordo el lobo? " + lobo.estasGordo())
    console.println("Esta saludable el lobo? " + lobo.estasSaludable())

}
```