

```

#!/usr/bin/env ruby
# encoding: utf-8
#

def media(vector)
  vector.inject(0.0) { |sx, x| sx += x } / vector.size
end

v = (1..100).to_a
v[88]=100
v[89]=100
v[90]=100
v[10]=1
v[11]=1
v[12]=1
v[18]=3
v[19]=3
v[20]=3
puts "A continuación se muestran los datos de prueba"
p v

print "El valor medio es igual a : "
p media(v)

def mediana(vector, clasificado=false)
  return nil if vector.empty?
  vector = vector.sort unless clasificado
  mitad = vector.size / 2
  return vector.size % 2 == 1? vector[mitad] : media(vector[mitad-1..mitad])
end

print "El valor de la mediana es igual a :"
puts mediana(v)

# La moda es el elemento que más se repite,
# si los elementos no se repiten, no existe.

def moda(vector, todas=true)
  histograma = vector.inject(Hash.new(0)) { |h, n| h[n] += 1; h}
  modas = nil
  histograma.each_pair do | item, times |
    modas << item if modas && times==modas[0] && todas
    modas = [times, item] if !modas && times > 1 or (modas && times>modas[0])
    # print modas, "\n"
  end
  return modas ? modas[1..modas.size] : modas
end

def moda(vector, todas=true)
  histograma = vector.inject(Hash.new(0)) { |h, n| h[n] += 1; h}
  modas = nil
  histograma.each_pair do | xi, ni |
    modas << xi if modas && ni==modas[0] && todas
    modas = [ni, xi] if !modas && ni > 1 or (modas && ni > modas[0])
    # print modas, "\n"
  end
end

```

```
    return modas ? modas[1...modas.size] : modas
end

print "A continuación se mostrarán los valores de la moda:"
print moda(v)
```