

```

1  class Circle
2    include Comparable
3    def initialize(r)
4      @radius = r
5    end
6    def area
7      Math::PI*@radius*@radius
8    end
9
10   def <=>(c)
11     if c.area == self.area then
12       0
13     elsif c.area > self.area then
14       1
15     else
16       (-1)
17     end
18   end
19   def to_s()
20     return self.area.to_s
21   end
22 end
23
24 class Shapes
25   include Enumerable
26   def initialize()
27     @instanceAry = []
28   end
29   def addCircle(c)
30     @instanceAry << c
31   end
32   def each()
33     @instanceAry.each{|ci| yield ci}
34   end
35 end
36
37
38 circle2 = Circle.new(2)
39 circle5 = Circle.new(5)
40 circle100 = Circle.new(100)
41 circle5b = Circle.new(5)
42
43
44 ta = (circle2<=>circle5)
45 tb = (circle100<=>circle5)
46 tc = (circle5<=>circle5b)
47 td = (circle5<=>circle5)
48
49 print "Following line should be 1 -1 0 0 to correlate with tests.\r\n"
50 print ta.to_s() + " " + tb.to_s() + " " + tc.to_s() + " " + td.to_s() + "\r\n"
51
52 s = Shapes.new()
53 0.upto(36) {
54   s.addCircle(Circle.new(rand(100)))
55 }
56 puts s.min()
57 puts s.sort()

```

```
[baseline]$ ruby ruby.rb
Following line should be 1 -1 0 0 to correlate with tests.
1 -1 0 0
30790.7495978336
30790.7495978336
30171.8558450764
27759.1126871194
23778.7147950211
23235.2192659501
22698.0069221863
21124.0690027378
20106.1929829747
19113.4497044403
18145.8391671346
13273.2289614169
11309.7335529233
10207.0345315132
9503.31777710913
8824.73376393373
7238.22947387088
6939.77817177985
6361.72512351933
5808.80481648753
5281.01725068444
5281.01725068444
5026.54824574367
4778.36242611007
4300.84034276443
2463.0086404144
2463.0086404144
1809.55736846772
1809.55736846772
907.92027688745
804.247719318987
452.38934211693
380.132711084365
314.159265358979
153.9380400259
28.2743338823081
12.5663706143592
3.14159265358979
[baseline]$
```