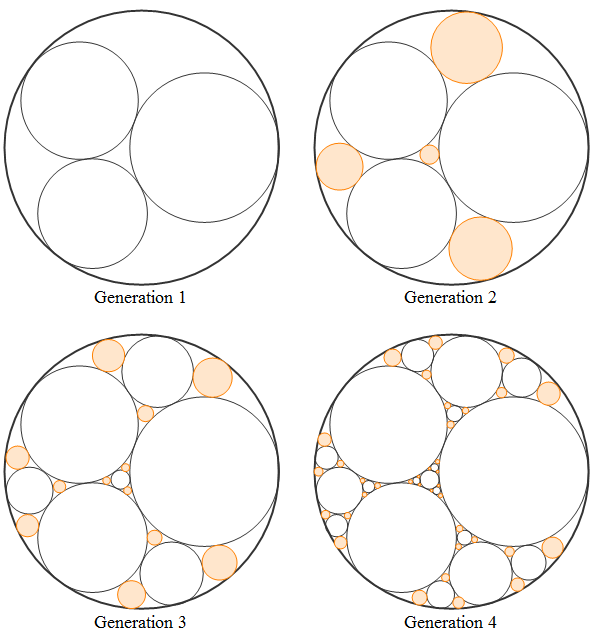
Integral circle packings 2

[Award] **10 pts**

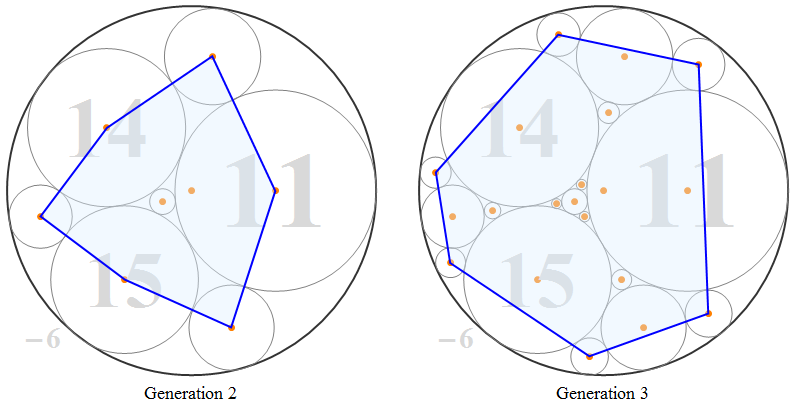
[Category] **Math**

See [Problem 457 – Integral circle packings 1](http://www.javaist.com/rosecode/show.php?no=457) for the concept of integral circle packing and quadruple.

According to the circles generated sequence in a circle packing, define the generation as below: the 4 mutually tangent circles are the first generation; fill the 4 interstices between them with 4 tangent circles forming the second generation; and repeatedly fill the interstices between mutually tangent circles of previous generation with further tangent circles forming the new generation. The circles of the first 4 generations are shown below. Actually the number of new generated circles in the *n*th (*n* >= 2) generation is 4×3n-2.



For a certain generation in a circle packing, a [convex hull](https://en.wikipedia.org/wiki/Convex_hull) can be formed from the set of center points of all circles. The figures below show the convex hull (blue polygon) of the 2nd and 3rd generation for integral circle packing with quadruple (-6, 11, 14, 15), respectively.



It can be calculated that the area of convex hull of the 2nd generation is 670/23023, and that of the 3rd generation is 45650065/866875849. In fact, the area of convex hull of any generation for any integral circle packing is a rational number.

Define *R*(*a*, *b*, *c*, *d*, *n*) be the ratio of the area of convex hull of the *n*th generation to the area of the bounding circle with quadruple (*a*, *b*, *c*, *d*). For example, R(-6, 11, 14, 15, 2) = (670/23023)/(pi/62) = 0.3334767170, and R(-6, 11, 14, 15, 3) = (45650065/866875849)/(pi/62) = 0.6034442099. Obviously the area ratio tends to 1 when *n* tends to infinity. Let *S*(t, n) be sum of all *R*(*a*, *b*, *c*, *d*, *n*) with **primitive root** quadruple (*a*, *b*, *c*, *d*) satisfying *d* <= *t*.

You are given *S*(10, 3) = *R*(-1, 2, 2, 3, 3) + *R*(-2, 3, 6, 7, 3) + *R*(-3, 5, 8, 8, 3) + *R*(-4, 8, 9, 9, 3) = 2.2572820988 (rounded to 10 digits after the decimal point).

Find *S*(50, 10). Give your answer rounded to 8 digits after the decimal point.

[Answer] **40.56007357**