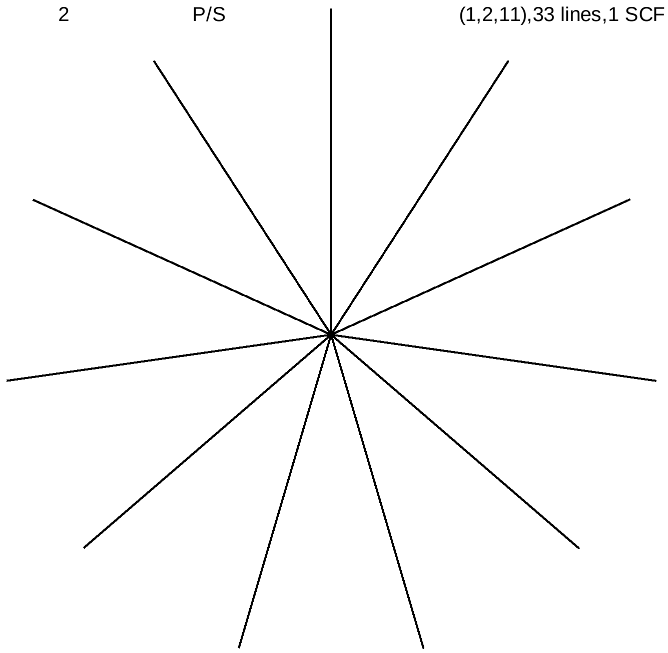
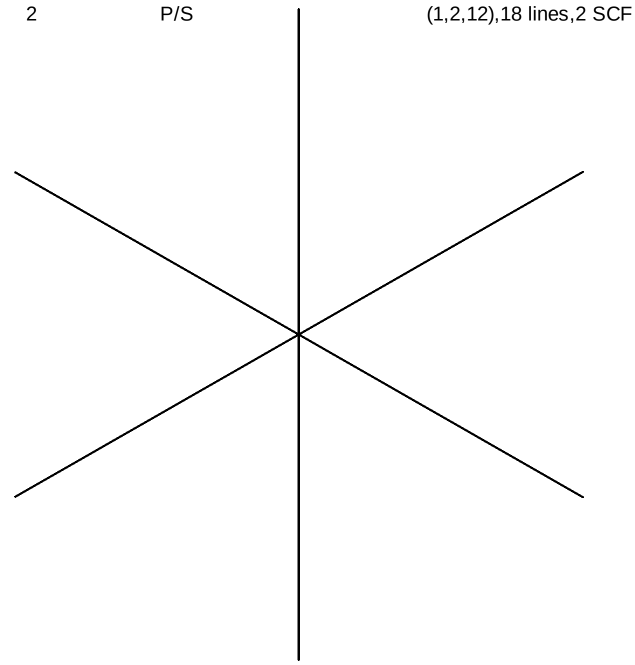
P = 2***S*** Images

An interesting pattern emerged in the ***P*** = ***m·S*** discussion where it was found that spiked images occur when ***P*** = 2***S***. When ***n*** is odd (like ***n*** = 11 at left), there are ***n*** spikes and when ***n*** is even (like ***n*** = 12 at right), there are ***n***/2 spikes.





The question, of course, is: Why?

We examine this by considering the general even and odd values of ***n***. Before turning to this question, consider the specific versions shown above. The first thing to recognize is that there are many more lines used than counted spikes in both models. It would not be surprising if the number of lines was twice the number of spikes since one needs to go into the center and back out to some other vertex, but in fact, it is three times this number (33 = 11·3 and 18 = 6·3). Second, although there appear to be 3 lines in the ***n*** = 12 image (3-*diameters*, from 12-6, 2-8, and 4-10 on the clockface), these three lines are actually 6-*radiuses*, one each to and from the center to hours 2, 4, 6, 8, 10 and 12.

Why are there 3× as many lines? The table shows the first 6 lines based on the ***P*** = 2***S*** jump pattern. Since our images are continuously drawn, the ending point of a line is the starting point of the next line. Six lines were shown to focus attention on the 3-line pattern that is shown twice: Lines 1 and 4 move to the center from a vertex; Lines 2 and 5 move out to *2 vertices* higher than the previous line's start (0 to 2 for Line 2 and 2 to 4 for Line 5); Lines 3 and 6 stay at the same vertex *because 2 jumps after the first time a vertex is used, it is reused*. This *cycle* repeats every 3 lines (even though the third is just a point).

		Vertex **number** of ***n***-gon or Center, **C**	
Line, ***L***	***P*** = 2***L****	Start	End
1	2	***0***	***C***
2	4	***C***	***2***
3	6	***2***	***2***
4	8	***2***	***C***
5	10	***C***	***4***
6	12	***4***	***4***

*Since ***P*** = 2***S*** produces the same image regardless of ***S***, set ***S*** = 1.

[**MA.** The values in the End column were obtained using this equation in *Excel*: =IF(2=MOD(P,3),"C",IF(1=MOD(P,3),(P+2)/3,P/3)) which simply operationalizes the jump pattern described here. The other equation used was to set Start in Line 2 =End in Line 1.]

A revised notion of a cycle. The first cycle in the traditional String Art model occurs once a polygonal vertex is achieved. The CPF model shows that a vertex can be achieved in two ways, 1) as the endpoint of a line from the previous vertex and 2) as an endpoint of a line from the center. The first way is at the start of a 3-jump set, the second is at the end of a set. The first cycle ends once a vertex is achieved as the end of a jump set so the cycle above is 3 lines long, not 2.

Even versus odd *n*. Another important attribute of this jump pattern is that it jumps vertices by 2 vertices each 3-jump set, whether ***n*** is even or odd. This is the attribute that creates half as many spikes when ***n*** is even.

n** is even.* If ***n is even, ***n*** = 2***k***. The ***k***th cycle ends at vertex 2***k*** = 0 and the circuit is complete with rays at all even vertices.

n** is odd.* If ***n is odd, ***n*** = 2***k***+1. The first ***k*** cycles create rays at even vertices ending at vertex 2***k*** = ***n***-1. The next cycle ends at vertex 1 = ***n***-1 + 2 - ***n***. The last ***k*** cycles create rays at odd vertices ending at 2***k***+1 = ***n*** completing the circuit.

NOTE: *The rest of this explainer has more mathematical detail, but you can simply ignore the equations if you do not want to wade through the detail and you can still obtain the gist of the argument.*

Background. Deeper into the table in the ***P*** = ***mS*** explainer (reproduced to the right), note that when ***m*** = 16 and ***n*** = 12, we once again have the 6-spike pattern examined above. A simple adjustment to the table on the previous page shows how the ***m*** = 16 image is created, and interestingly, it is quite different from how the ***m*** = 2 image was created.

m	***n*** = 11 Image*	***n*** = 11 3***n-m***	***n*** = 12 Image	***n*** = 12 3***n-m***
1	VF = 11 RP	32	VF = 12 RP	35
2	11 R	31	6 R	34
3	11 P	30	12 P	33
4	11,2 RS	29	**3 T v.1**	32
5	VF = 11 RP	28	VF = 12 RP	31
6	11,2 S	27	Hexagon	30
7	11,3 RS	26	12,3 RS	29
8	11,2 RS	25	**3 T v.2**	28
9	11,3 S	24	Square	27
10	11,4 RS	23	6,2 RS	26
11	**Obtuse Δ**	22	12,3 RS	25
12	11,4 S	21	Triangle	24
13	**11,5 RS**	20	**12,5 RS**	23
14	11,4 RS	19	6,2 RS	22
15	11,5 S	18	12,5 S	21
16	11,5 RS	17	6 R	20
17			12,5 RS	19
18			Vertical	18

*Acronyms: #,## ***n*** ,***J***-star; P-polygon; R-rays; RS-star with rays; S-star; T-equilateral triangle; VF-Vertex Frame. **BOLD images are shown below**.

The ***m*** = 16 image is shown at bottom right. It is the SAME static image as ***m*** = 2. Recall that ***n*** = 12 so we can think of vertices as hours on the clockface.

Adjusting the table on the previous page. Given the automation of Start and End points in the **MA** note on the prior page, all that needs to be done is to change ***P*** = 2***L*** to ***P*** = 16***L*** in the second column. This is done by typing 16 in B1 and then referencing it to create the equation in cell B3 (the equation is =B1*A3). Drag this equation to obtain multiples of 16 for ***P***. The results are shown in the first 4 columns of the table below.

The first two lines. Rather than focus on the entire table (which has more rows AND columns), initially focus on the first two lines because they show that a very different pattern from the ***m*** = 2 pattern on the prior page. The first line is the vertical diameter, going from 0 (or 12 o'clock) to 6 and the second is the vertical radius from 6 to the center. The second line backtracks half of the first line.

The first line does end in a vertex, but it is the end of the *Jump* from vertex 5 to vertex 6 using the jump-set pattern of *Jump, In, Out*. As such, it is not the end of the first cycle (as discussed above) since it is not on a move out from the center back *Out* to a vertex.

The rest of the table. The third line ends at vertex 16, but of course, that is the same as vertex 4 (since ***n*** = 12) so it is a jump out from the center to vertex 4. This is the end of the first cycle. Rather than look at larger and larger values for Start and End vertex numbers, the vertices are transformed in columns 5 and 6 using the equations noted at the bottom of the table via the MOD function (the *Excel* COUNT function produces a 1 if a number is present). The three line cycle is diameter, radius, radius as noted in the final column. Each cycle creates 1/3rd of the image (unlike ***m*** = 2 which is 1/6th). The table was extended to show all 9 lines in the image and a line was included to note the end of each cycle.

9 lines versus 18 lines. The same image is created in both cases, but it takes half as many lines with ***m*** = 16 as ***m*** = 2. But remember that six of the lines for ***m*** = 2 were actually points rather than lines. If you think of it wholistically each of the three lines in the ***m*** = 16 version includes three lines, a diameter, and both radiuses, they just are mapped in different cycles (e.g., lines 3, 4, and 5 map the 4-10 line).

m = **16**		Vertex **number** of ***n***-gon or Center, **C**		Vertices (MOD 12)		Diameter
Line, ***L***	***P*** = ***mL****	Start	End	Start	End	or radius
1	16	***0***	***6***	**0**	**6**	D
2	32	***6***	***C***	**6**	**C**	R
3	48	***C***	***16***	**C**	**4**	R
4	64	***16***	***22***	**4**	**10**	D
5	80	***22***	***C***	**10**	**C**	R
6	96	***C***	***32***	**C**	**8**	R
7	112	***32***	***38***	**8**	**2**	D
8	128	***38***	***C***	**2**	**C**	R
9	144	***C***	***48***	**C**	**0**	R
End		IF(2=MOD(P,3),"C",IF(1=MOD(P,3),(P+2)/3,P/3))				
MOD 12 Start		IF(COUNT(Start)=1,MOD(Start,12),"C")				
MOD 12 End		IF(COUNT(End)=1,MOD(End,12),"C")