Turn	1	2	3	4	5	6	7	8	9	10	11	12	...
k	1	2	1	3	1	2	1	4	1	2	1	3	...

distance $= \frac{n}{2^k}$ where n = total vertices

Notice $k=1$ is at every other turn;
$k=2$ is at every other of the remaing turns;
$k=3$ is at every other of the now remaing turns;
and so on...

Thus for $\forall m \in \mathbb{N}$ (including 0),

$k=1$ is at $1+2m$ turns

$k=2$ is at $2+4m$ turns

$k=3$ is at $4+8m$ turns

...

$k=a$ is at $2^{a-1}+2^a m$ turns

How the algorithm works:

- Initialize an empty array of size $n-1$ because that's how many turns we take to explore every vertex.

- Start with $k=1$

~~1~~ 2 ~~3~~ 4 ~~5~~ 6 ~~7~~ 8 ~~9~~ 10 ~~11~~ 12 ...

These turns (positions in the output array) $= \frac{n}{2}$

note: position = turn -1 b/c 0-indexed

Now we do $k=2$

$\cancel{1}\ \cancel{2}\ \cancel{3}\ 4\ \cancel{5}\ \cancel{6}\ \cancel{7}\ 8\ \cancel{9}\ \cancel{10}\ \cancel{11}\ 12 \ldots$

And so on, until $2^k = n$, which will fill in the last position at $\frac{n}{2}$.