Last One Standing

STUDENTS: Alex Florin Oprea, Adrian Solcanu, Victor Solcanu, Alexandru Tudor Ionita, Andrei Razvan Damoc, Ana Tudora (10th grade)

SCHOOL: *'Costache Negruzzi'* National College, Iasi, Romania

TEACHERS: Adrian Zanoschi, *'Costache Negruzzi'* National College, Iasi, Romania; Vlad Tuchilus, student, The Queen's College, University of Oxford

RESEARCHER: Vlad Tuchilus, student, The Queen's College, University of Oxford

February 2025

Abstract

1 Introduction

In the game show *'Last One Standing'*, n contestants are arranged in a circle, each numbered from 1 to n. The host eliminates contestant number 1, and then counts every second contestant clockwise, who is then asked to leave the circle. This process repeats until only one contestant remains.

If you are a contestant on this game show, what seat should you sit in to be the last one standing?

Further possible questions are:

- If instead every k^{th} contestant is eliminated, what is the optimal seat?
- Further, suppose the m^{th} contestant picked is the winner. What is the optimal seat in this game?
- Suppose the host now alternates, eliminating every second player, then every third player, and then repeating. What should your strategy be now?

- Studying other arrangements of seats such as a figure 8 consisting of two circles that share a central contestant.

For any game with n participants, the answer can be found simply by actual simulation. However, this can take a very long time and is a mistake-prone method for large values of n. We can see that this type of algorithm has complexity $O(n)$, as it takes n steps to complete.

Our goal is to find more efficient algorithms for various games and the way to achieve this is by using one or more recurrence relations. However, not all sets of recurrence relations are better. For example, a set that would allow us to compute the winning position from knowing the winning position for a game with just 1 fewer player wouldn't improve the complexity, as approximately n iterations would still be necessary to find the answer knowing what the winning position in a 2-player game is. The number of steps "jumped" lower with each iteration mustn't be constant, but at least directly proportional to n. In this situation, we would obtain algorithms of logarithmic complexity $O(\log(n))$, which is exactly what we have managed. In figure 1 we can see the huge difference between the number of necessary iterations for an algorithm of linear complexity compared to one of logarithmic complexity.

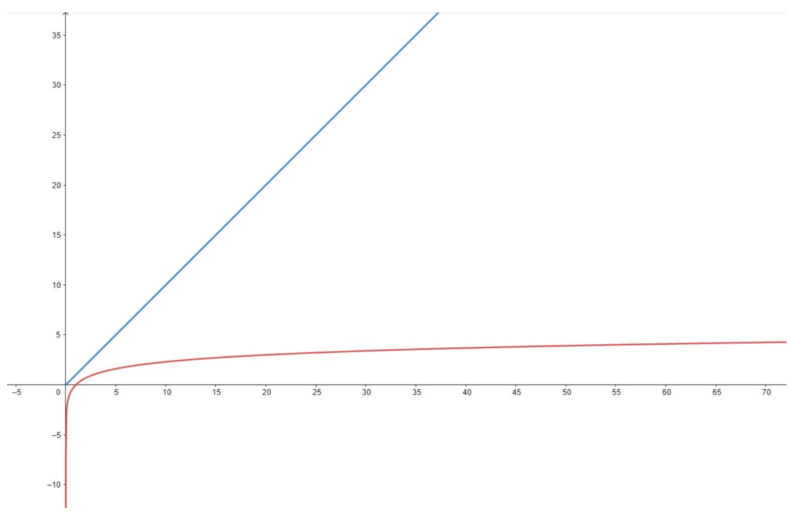

Figure 1: $O(n)$ VS $O(\log(n))$

With this approach, in subsection 2.1, we have completely solved the *'Last One Standing'* game described above, which in the rest of the article shall be referred to as the basic game. For this case, the formula obtained has an even simpler interpretation using binary representations, which we have presented in subsection 2.2. In the following 3 subsections, we work on a game similar to the basic one, with the only common change being that the number of contestants "jumped over" at each step is no longer 2, but k, any integer greater than or equal to 2. We first find the solution, then change the starting point, then change the number of rounds played. Further on, in subsection 2.6, we cover the game in which k is no longer fixed, but alternates between 2 and 3, and subsection 2.7 solves the basic game played on a different arrangement of seats. We conclude in section 3.

2 Our results

For easier understanding, from now on we will denote the seated contestants by numbers, e.g. for a game with n people seated in a circle, we will work with the numbers $1, 2, ..., n$ on a circle. Also, let $f(n)$ generally denote the optimal seat for an arrangement with n contestants. Occasionally, for generalizations, f might have multiple arguments.

2.1 Finding recurrences for the basic game

The nature of the game brought to mind the idea of recurrence relations, which we are going to make use of. Clearly, $f(1) = 1$. For $n \geq 2$, due to the importance of the number 2, the game's step size, we will consider 2 cases depending on the parity of n.

1. n is even, or equivalently, $n = 2x$, where x is a positive integer

The first number eliminated is 1. Before judging number 2, a judgment of each number will have been made. At this point, all the odd numbers are eliminated. The remaining numbers form a circle with x participants: $\{2, 4, 6, .., 2x\}$, out of which 2 is eliminated first. Since all the remaining numbers are even, we can divide them by 2, leading us to an equivalent game: the basic game for $x = \frac{n}{2}$. We can now deduce the following recurrence relation: $f(2x) = 2f(x)$.

2. n is odd, or equivalently, $n = 2x+1$, where x is a positive integer

Our second case is extremely similar to what we wrote above. In the first full rotation, all odd numbers are eliminated. This leaves us with a circle with x contestants that are numbered $2, 4, ..., 2x$. However, despite all these similarities, the game is not the same, and that is because the next number eliminated is 4, not 2. To make this situation easier to follow and understand, we can subtract 2 from all the numbers. Now the contestants have the following numbers: $0, 2, ..., 2x - 2$. Since $2x$ does not feature in our list, we can rename 0 to $2x$. If we now divide all numbers by 2, it leads us to a basic game for x numbers. Everything leads to the following recurrence relation: $f(2x + 1) = (2f(x) + 2)$.

Note that in the formula above if $f(x) = x$, then $f(2x + 1) = 2x + 2 > 2x$, which is not possible. In this case, we see that, because of the circular configuration and the elimination of 1, we have $f(2x + 1) = 2$. Otherwise, the recurrence relation remains. Thus, the recurrence relation reads

$$f(2x+1) = \begin{cases} 2, \text{ if } f(x) = x \\ 2f(x) + 2, \text{ otherwise} \end{cases}, \forall x \geq 1.$$

2.2 The general formula for the basic game and its (re)interpretation

After finding the answer in the form of recurrence relations, we decided to simplify it by determining a general formula (depending only on n).

The formula to be proven is $f(n) = 2\left(n - 2^{\lfloor \log_2(n-1) \rfloor}\right)$. We use the following formulas, which were proved in the previous subsection:

(1) $f(2x) = 2f(x)$, for any positive integer x

(2) $f(2x+1) = 2f(x) + 2$, for any positive integer x with $f(x) \leq x - 1$

(3) $f(2x+1) = 2$, for any positive integer x with $f(x) = x$

We break the proof down into 3 parts:

I. We prove that $n = 2^k$ if and only if $f(n) = n$.

Suppose that $n = 2^k$. We know $f(1) = 1$. Using simple induction and (1), we get that $f(2^k) = 2^k$ for any non-negative integer k, and, therefore, $f(n) = n$.

Conversely, suppose that $f(n) = n$ and n is not a power of 2. Then $n = 2^a(2b+1)$, for a non-negative integer a and a strictly positive integer b. Since $f(n)$ must be even, this means n is even and $a \geq 1$. Using (1) repeatedly, we find that $f(2b+1) = 2b+1$. Since f only takes even values for numbers greater than 1, we reach a contradiction. Therefore, our assumption is wrong and n must be a power of 2.

II. We prove that $n = 2^k + 1$ if and only if $f(n) = 2$, for $n \geq 3$.

Suppose that $n = 2^k + 1$. Using (3) and what we proved in I., we find that $f(n) = 2$.

Now assume $f(n) = 2$. If $n = 2p$, then, by (1), $f(p) = 1$ and $p \geq 2$, a contradiction. If $n = 2p + 1$ and it is the case to apply (2), we get $f(p) = 0$, again a contradiction. Then, (3) must be applied, meaning $f(p) = p$, which, by I., is equivalent to p being a power of 2. We reach the conclusion.

One can easily check then that in the two cases above, our formula holds.

III. n is not a power of 2, neither is $n - 1$, and $n \geq 6$ (smaller cases have already been covered)

Let $P(n)$ be the statement that $f(n) = 2\left(n - 2^{\lfloor \log_2(n-1) \rfloor}\right)$ for integer-valued $n \geq 6$. Suppose $P(n)$ is true for any n with values between $2^k + 2$ and $2^{k+1} - 1$ inclusively.

We write for $2^k + 1 \leq n \leq 2^{k+1} - 1$, $f(2n) = 2f(n) = 2 \cdot 2(n - 2^{\lfloor \log_2(n-1) \rfloor}) = 2(2n - 2^{\lfloor \log_2(n-1) \rfloor + 1}) = 2(2n - 2^{\lfloor \log_2(n-1)+1 \rfloor}) = 2(2n - 2^{\lfloor \log_2(2n-2) \rfloor})$. Since $2n - 2$ is even and $2n - 1$ is odd, $\lfloor \log_2(2n-2) \rfloor = \lfloor \log_2(2n-1) \rfloor$ and we have proved $P(2n)$.

We do the same for $2n + 1$. The use of (2) and not (3) is forced by the fact that n is not a power of 2:

$f(2n+1) = 2f(n) + 2 = 2 \cdot 2(n - 2^{\lfloor \log_2(n-1) \rfloor}) + 2 = 2(2n - 2^{\lfloor \log_2(n-1) \rfloor + 1} + 1) = 2((2n+1) - 2^{\lfloor \log_2(n-1)+1 \rfloor}) = 2((2n+1) - 2^{\lfloor \log_2(2n-2) \rfloor}$.

We use the fact that n is not a power of 2, to reach that $\lfloor \log_2(2n-2) \rfloor$ and $\lfloor \log_2(2n) \rfloor$ must be equal. We have now proved $P(2n+1)$.

Thus $P(n)$ also holds for $2^{k+1}+2 \leq n \leq 2^{k+2}-1$. With the induction step now proven, we can affirm that the formula holds for any positive integer n.

To transform the formula into something easier to write, we had the idea of reconstructing it into binary, so that $\log_2$ can be easily interpreted. Doing so, $n-1$, where n is the number of participants, can be written as $\overline{1x}$, where x is a binary number with d digits. Because of the fact that 2^d is equal to $\overline{100...0}_{(2)}$, the number of digits being $d+1$, the function can be written as such: $f(n) = 2(\overline{1x}+1-2^d)$, which can be further simplified as $2(x+1)_{(2)}$, which in the end is $\overline{x0}_{(2)}+2$. Although harder to compute, this recalculated formula is much easier to write than its base 10 counterpart.

2.3 Finding formulas for a general k

In this section, we consider the game in which every k^{th} contestant ($k \geq 2$) gets eliminated, starting from 1. This generalises the two previous sections, for which $k = 2$.

For our proof, we will use the following auxiliary functions:

Definition 2.1. *Let Φ_w be a function defined on the set of strictly positive integers with values in the set $1, 2, ..., w$ such that $\Phi_w(v) = v - w \cdot \lfloor \frac{v-1}{w} \rfloor$.*

Definition 2.2. *Let Θ_w be a function defined on the set of strictly positive integers with values in the same set such that $\Theta_w(v) = v + \lfloor \frac{v-1}{w-1} \rfloor + 1$.*

The function Φ_w returns the residue of a number modulo w, replacing 0 with w. This helps when the counting during the game passes over the last player and ensures 0 does not show up in our results.

The function Θ_w helps us "jump" over the already eliminated players when applying the function.

Next up we will consider the quotient and residue of n upon division with k. Let these two be a and b, respectively, where a is a nonnegative integer and b belongs to the set $0, 1, ..., k-1$.

In the first case, $b = 0$ and $n = a \cdot k$. We eliminate the first a numbers, namely $1, k+1, ..., (a-1) \cdot k+1$. We are left with a smaller game starting with $a \cdot (k-1)$ numbers, for which we know the answer is the $f((a-1) \cdot k)^{\text{th}}$ number, up to a relabeling. The first one to be eliminated is 2. Note that here Θ_k is a bijection between $\{1, \ldots, a \cdot (k-1)\}$, the set of contestants for the smaller game, and the $n-a$ numbers left on the board $\{ck+d \mid 0 \leq c \leq a-1, 2 \leq d \leq k\}$ in our larger game. It allows us hence to convert the answer for the smaller game to the larger one via the appropriate relabeling. Therefore, $f(a \cdot k) = \Theta_k(f(a \cdot (k-1)))$.

In the second case, $b = 1$ and $n = a \cdot k+1$. The first time around, we take the first $a+1$ steps, eliminating $1, k+1, ..., a \cdot k+1$. The rest of the game is played starting with $a \cdot (k-1)$ numbers, out of which the first eliminated one is $k+2$. We shall find the answer using the formula $f(a \cdot k+1) = \Phi_n(\Theta_k(f(a \cdot (k-1))) + k)$. One can check that the function $g : x \mapsto \Phi_n(\Theta_k(x) + k)$ is a bijection between the possible optimal seats' position and their actual label in the game, and that $g(x)$ represents the x^{th} number on the board if we start at $k+2$ and go clockwise.

The final case is for a general b taking one of the values $2, 3, ..., k-1$. We begin with the first $a+1$ eliminations. We are left with $a \cdot (k-1) + (b-1)$ numbers on the board, the next to be eliminated being $k-b+2$, which is the k^{th} remaining number after $a \cdot k + 1$. Now, because of the inconvenience of carrying over 1, we distinguish 3 situations based on the value of $f(a \cdot (k-1) + (b-1))$: (We should mention that the answer cannot be 1 unless $a(k-1) + (b-1) = 1$, in which case, since $b \geq 2$, k must be zero which obviously does not correspond to an actual game that we could play)

1. if it is between 2 and $b-1$ (inclusively), the formula will be $f(n) = f(a \cdot (k-1) + (b-1)) + k - b + 1$.

2. if it lies between b and $a \cdot (k-1) + 2b - k - 1$ (inclusively), the formula is $f(n) = \Theta_k(f(a \cdot (k-1) + (b-1)) - (b-1)) + k$.

3. if it lies between $a \cdot (k-1) + 2b - k$ and $a \cdot (k-1) + (b-1)$ (inclusively), the formula will be $f(n) = f(a \cdot (k-1) + (b-1)) - (a \cdot (k-1) + 2b - k) + 2$.

What all these formula do is that they map $f(a \cdot (k-1) + (b-1))$ (which is the position of the optimal seat) to the $f(a \cdot (k-1) + (b-1))^{\text{th}}$ number left on the board if we start at $k - b + 2$ and go clockwise.

These formulas, together, allow us to find $f(n)$ in a logarithmic number of steps i.e. with $O(\log(n))$ complexity.

Since we have completed the initial version of the game, in the rest of this article we now try to find answers to questions that impose a slight change in rules.

2.4 Generalization on the first player eliminated

The game seems to be unfair, in the way that contestant 1 can never win. Therefore, we decided to change the first contestant to be eliminated. So, let $s \in \{1, 2, ..., n\}$ be the first eliminated contestant. By subtracting $s-1$ from all numbers, we "rotate" the circle in order to take s to 1's previous position. Using the rotation argument, we can say that $f(n,s) = f(n,1) + (s-1)$, where $f(n,s)$ denotes the optimal seat for n people, out of which the s^{th} one is eliminated first. We notice this formula is not accurate, since $f(n,1) + (s-1)$ might exceed n. In order to solve this, we use the function Φ_n which we defined earlier on. In the end, we get the following formula:

$$f(n,s) = \Phi_n(f(n,1) + (s-1)).$$

Hence, to compute $f(n,s)$, we can use the $O(\log(n))$ algorithms described in the previous sections to compute $f(n,1)$, and then use the formula above.

2.5 What happens if the winner is picked after m rounds?

In this subsection, we consider an iteration of the original game in which the winner is the m^{th} player eliminated, where $m \leq n$ is some positive integer. For the basic game of course $m = n$. For simplicity, we again eliminate 1 first, but our results can be easily adapted to starting at some player s as in subsection 2.4. We write here $f(n, m)$ for the number of the winning player in this game.

Right after we started solving this problem, we realised that our main concern will be the situation when the winner is not picked in the first full rotation, because we are left with numbers missing from the circle. In order to figure out how we should approach this new game, we analysed the case where $k = 2$. Here we have another two subcases ($n = 2x$ and $n = 2x + 1$), but they are almost identical.

We will start with $n = 2x$. If $1 \leq m \leq x$, the game does not reach a full rotation, so the answer is $f(n, m) = 2m - 1$. If $m > x$, we only have the even numbers left for the second rotation, with number 2 being the first eliminated. This way, we can divide each contestant number by 2, which is equivalent to starting a game with x contestants from scratch. Keeping in mind the previous rounds and our division trick, the answer is $2f(x, m - x)$.

If $n = 2x + 1$, the first rotation will include $x + 1$ rounds and the second rotation will be equivalent to a game with x contestants. Here, the answer is $f(n, m) = 2m - 1$ if $m \leq x + 1$, and $f(n, m) = 2f(x, m - x - 1)$ otherwise.

Now it is time to analyse the problem for a general $k \geq 2$. In order to cover all possible outcomes of this case, we can write n as $a \cdot k + b$ as before. No matter the value of b, if $m \in \{1, 2, ..., a\}$, the answer we are looking for is $f(n, m) = (m - 1)k + 1$. For $m > a$, we need to look at another three subcases: $b = 0$, $b = 1$ and $b > 1$.

If $b = 0$, the first contestant eliminated is 1, which can be written as $(1 - 1) \cdot k + 1$. The next contestant eliminated is $k + 1$, which is written as $(2 - 1) \cdot k + 1$. As this process continues, the last contestant eliminated before a full rotation is completed is $(a - 1) \cdot k + 1$. We can see that in the second rotation there are $n - a$ contestants left and $m - a$ rounds remaining. The answer to this situation can be found with the help of the function Θ_k that has been described earlier in this article. Therefore, we can write the answer in the form of $\Theta_k(f(n - a, m - a))$.

If $b = 1$, the reasoning is similar to the case above. Here, the last contestant eliminated in the first full rotation is n. This makes the circle remain with $n - a - 1$ contestants and $m - a - 1$ rounds still to go. The answer can be written in the form of $\Phi_n(\Theta_k(f(n - a - 1, m - a - 1)) + k)$.

If $b > 1$, the last contestant eliminated in the first rotation is $a \cdot k + 1$. Now, in the circle, there are $m - a - 1$ rounds to go and $a \cdot (k - 1) + b - 1$ contestants remaining. The first contestant to start the new rotation is $k - b + 2$. We can now write the final result by using our auxiliary function Θ, in the following manner, depending on the value of $f(a \cdot (k - 1) + b - 1, m - a - 1)$ as follows:

1. if it is between 2 and $b - 1$ (inclusively), the formula will be $f(n) = f(a \cdot (k - 1) + (b - 1), m - a - 1) + k - b + 1$.

2. if it lies between b and $a \cdot (k-1) + 2b - k - 1$ (inclusively), the formula is $f(n) = \Theta_k(f(a \cdot (k-1) + (b-1), m - a - 1) - (b-1)) + k$.

3. if it lies between $a \cdot (k-1) + 2b - k$ and $a \cdot (k-1) + (b-1)$ (inclusively), the formula will be $f(n) = f(a \cdot (k-1) + (b-1), m - a - 1) - (a \cdot (k-1) + 2b - k) + 2$.

The reasoning for these formulas is precisely analogous to subsection 2.3, the only difference being that we now account for the remaining number of steps m.

2.6 Generalization on a varying k

In this subsection we are going to analyse the situation in which k is not fixed, but alternates between 2 and 3. To find suitable recurrence relations, we will consider the residue of n upon division with $5 = 2 + 3$. As before, we will deal with the problem of already eliminated numbers using an auxiliary function.

Definition 2.3. *Let Ψ be a function defined on the set of positive integers onto itself. For an arbitrary positive integer x, there exist unique positive integers y and r, with $r \in \{1, 2, 3\}$ and $x = 3 \cdot y + r$. Then, we define $\Psi(x) = 5 \cdot y + 2$, if $r = 1$, $\Psi(x) = 5 \cdot y + 4$, if $r = 2$, or $\Psi(x) = 5 \cdot y + 5$, if $r = 3$.*

In each case, Ψ is used to map the result of a smaller game, to our larger game, taking into account the fact that the remaining players in the larger game no longer have consecutive numbers, but only the ones congruent to either 0, 2, or 4 modulo 5. For example, in the first case $\Psi(x)$ gives the number of the x^{th} player remaining on the board starting at player 2 and going clockwise.

Due to symmetry and the variation of k, we will define the function g, which represents the answer to a game played similarly, where k alternates between 2 and 3, but in which k first takes on the value of 3. This way, we will be able to find simpler to write and use formulas, that allow us to compute both f and g at the same time.

If $n = 5a$, in the first rotation, we eliminate the numbers $5 \cdot i + 1$ and $5 \cdot i + 3$, for $i = \overline{0, a-1}$. There remain $5 \cdot a - 2 \cdot a = 3 \cdot a$ players in the game, of which 2 is the first one to be eliminated. Therefore, $f(5 \cdot a) = \Psi(f(3 \cdot a))$.

If $n = 5a + 1$, in the first rotation, we eliminate the numbers $5 \cdot i + 1$ and $5 \cdot i + 3$, for $i = \overline{0, a-1}$, and $5 \cdot a + 1$. There remain $5 \cdot a + 1 - 2 \cdot a - 1 = 3 \cdot a$ players in the game, of which 4 is the first one to be eliminated and, after it, the next "unlucky" number is the one 3 places further, meaning the rest of the game is a g-function kind of game. Therefore, $f(5 \cdot a + 1) = \Psi(g(3 \cdot a) + 1)$, unless there is "jump" over $5a$. This would only happen if $g(3 \cdot a) = 3 \cdot a$ and would mean the answer is 2. Otherwise, the formula stated above holds.

If $n = 5a + 2$, in the first rotation, we eliminate the numbers $5 \cdot i + 1$ and $5 \cdot i + 3$, for $i = \overline{0, a-1}$ and $5 \cdot a + 1$. There remain $5 \cdot a + 2 - 2 \cdot a - 1 = 3 \cdot a + 1$ players in the game, of which 2 is the first one to be eliminated. Afterwards, 3 places are to be jumped, meaning the rest of the game fits to the conditions of g. Therefore, $f(5 \cdot a + 2) = \Psi(g(3 \cdot a + 1))$. As opposed to the previous case, there will be no possible "jump" over $5a + 2$, since all numbers ahead of 2 have already been crossed out.

If $n = 5a + 3$, in the first rotation, we eliminate the numbers $5 \cdot i + 1$ and $5 \cdot i + 3$, for $i = \overline{0, a}$. There remain $5 \cdot a + 3 - 2 \cdot (a + 1) = 3 \cdot a + 1$ players in the game, of which 5 is the first one to be eliminated, and then another number 2 places further, meaning the game is of the f-function kind. Therefore, $f(5 \cdot a + 3) = \Psi(f(3 \cdot a + 1) + 2)$, unless there is a "jump" over $5a + 2$. This would happen if $f(3 \cdot a + 1) = 3 \cdot a$, which would give the answer 2, or if $f(3 \cdot a + 1) = 3 \cdot a + 1$, which would give the answer 4. Otherwise, the formula holds.

If $n = 5a + 4$, in the first rotation, we eliminate the numbers $5 \cdot i + 1$ and $5 \cdot i + 3$, for $i = \overline{0, a}$. There remain $5 \cdot a + 4 - 2 \cdot (a + 1) = 3 \cdot a + 2$ players in the game, of which 4 is the first one to be eliminated, after which follows another number 2 places further, meaning the rest of the game if of the f-function kind. Therefore, $f(5 \cdot a + 4) = \Psi(f(3 \cdot a + 2) + 1)$. This formula holds, unless $f(3 \cdot a + 2) = 3 \cdot a + 2$, which would mean there is a "jump" over $5 \cdot a + 4$, and give the answer 2.

What we have left to do now, in order to finish the set of recurrence relations, is to find a set similar to the one above, but for g. Again, we will break the analysis into 5 parts, depending on the residue of n through division by 5. For this part, we will need another auxiliary function, defined very similarly to Ψ and with the same purpose.

Definition 2.4. *Let Ψ' be a function defined on the set of positive integers onto itself. For an arbitrary positive integer x, there exist unique positive integers y and r, with $r \in \{1, 2, 3\}$ and $x = 3 \cdot y + r$. Then, we define $\Psi(x) = 5 \cdot y + 2$, if $r = 1$, $\Psi(x) = 5 \cdot y + 3$, if $r = 2$, or $\Psi(x) = 5 \cdot y + 5$, if $r = 3$.*

If $n = 5 \cdot a$, in the first rotation, the numbers crossed out are $5 \cdot i + 1$ and $5 \cdot i + 4$, for i taking all the values from 0 to $a - 1$, meaning there are $5 \cdot a - 2 \cdot a = 3 \cdot a$ numbers left on the board, of which 2 is the first one to be eliminated and, afterwards, another number 3 places further, the rest of the game being of the g-function kind. Therefore, since "jumps" are not possible, $g(5 \cdot a) = \Psi'(g(3 \cdot a))$, for any positive integer a.

If $n = 5 \cdot a + 1$, in the first rotation, the numbers crossed out are $5 \cdot i + 1$ and $5 \cdot i + 4$, for $i = \overline{0, a - 1}$ and $5 \cdot a + 1$, leaving in the game $5 \cdot a + 1 - 3 \cdot a - 1 = 3 \cdot a$ numbers, out of which 5 is the first one to be eliminated and, after that, another one 2 places further, meaning the rest of the game is of the f-function kind. If $f(3 \cdot a) \leq 3 \cdot a - 2$, then $g(5 \cdot a + 1) = \Psi'(f(3 \cdot a) + 2)$. Otherwise, if $f(3 \cdot a) = 3 \cdot a - 1$, the answer is 2 and, if $f(3 \cdot a) = 3 \cdot a$, the answer is 3.

If $n = 5 \cdot a + 2$, the numbers crossed out during the first rotation are $5 \cdot a + 1$ and $5 \cdot a + 4$, for $i = \overline{0, a - 1}$ and $5 \cdot a + 1$, leaving $5 \cdot a + 2 - 3 \cdot a - 1 = 3 \cdot a + 1$ numbers on the table, out of which 3 is the first one to be eliminated, and then another number 2 places further. This means the rest of the game is of the f-function kind. If $f(3 \cdot a + 1) \leq 3 \cdot a$, which would mean there is no "jump" over $5 \cdot a + 2$, the formula is $g(5 \cdot a + 2) = \Psi'(f(3 \cdot a + 1) + 1)$. Otherwise, the answer is 2.

If $n = 5 \cdot a + 3$, the numbers crossed out during the first rotation are $5 \cdot a + 1$ and $5 \cdot a + 4$, for $i = \overline{0, a - 1}$ and $5 \cdot a + 1$, leaving $5 \cdot a + 3 - 3 \cdot a - 1 = 3 \cdot a + 2$

numbers on the table, out of which 2 is the first one to be eliminated and then another one 2 places further. Since "jumps" are not possible, the formula that holds for any positive integer a is $g(5 \cdot a + 3) = \Psi'(f(3 \cdot a + 2))$.

If $n = 5 \cdot a + 4$, the numbers crossed out during the first rotation are $5 \cdot a + 1$ and $5 \cdot a + 4$, for $i = \overline{0, a}$, leaving $5 \cdot a + 4 - 2 \cdot (a + 1) = 3 \cdot a + 2$ number on the table, out of which 3 is the first on to be eliminated and, after that, another number 3 places further, meaning the rest of the the game is of the g-function kind. Therefore, $g(5 \cdot a + 4) = \Psi'(g(3 \cdot a + 2) + 1)$, if $g(3 \cdot a + 2) \leq 3 \cdot a + 1$ and there are no "jumps" over $5 \cdot a + 3$. Otherwise, the answer is 2.

Combining all the formulas obtained above, the values of both f and g can be calculated in a logarithmic number of steps.

2.7 Changing shape

In the final part of our article, we are going to use the rules of the initial game (so $k = 2$), only tweaking the shape in which the players are arranged. Instead of a circle, we will consider that the participants form two circles that share a common person, a shape similar to the digit 8. Therefore, the number of participants will be odd. For ease of notation, we will number the participants in the following manner: out of $2n + 1$ people, the middle one is number $n + 1$, the ones in the top circle are numbered 1 through n in clockwise order, while the ones in the bottom circle are numbered $n + 2$ through $2n + 1$ in anti-clockwise order. Let $F(2n + 1)$ denote the answer, the position of the last eliminated participant.

We will have two recurrence relations based on the number of participants' remainder when divided by 4.

If there are $2n + 1 = 4x + 1$ participants, after going through all of them, the common person, number $2x + 1$ will have been eliminated. Therefore, we end up with $2k$ participants numbered 2 through $4x$, arranged in a circle and out of which the first one to be eliminated is 4. Using other results obtained earlier in this article, we get $F(4x + 1) = 2 \cdot \Phi_{2x}(f(2x) + 1)$. This part of the solution is represented in figure 2.

If there are $2n + 1 = 4x + 3$ participants, after going through all of them, the common person, number $2x + 2$ will not have been eliminated. Therefore, we end up with a similar figure, formed by two (smaller) circles of equal sizes and a common person, out of which the one next to the common participant is the first one to be eliminated. In fact, we now have $2x + 1$ players, and the only difference between our game and playing a figure 8 game with $2x + 1$ contestants is that all our labels are doubled. Thus, we find $F(4x + 3) = 2 \cdot F(2x + 1)$.

With these formulas, we have solved the case of the different arrangement and can now find the values of F in a logarithmic number of steps.

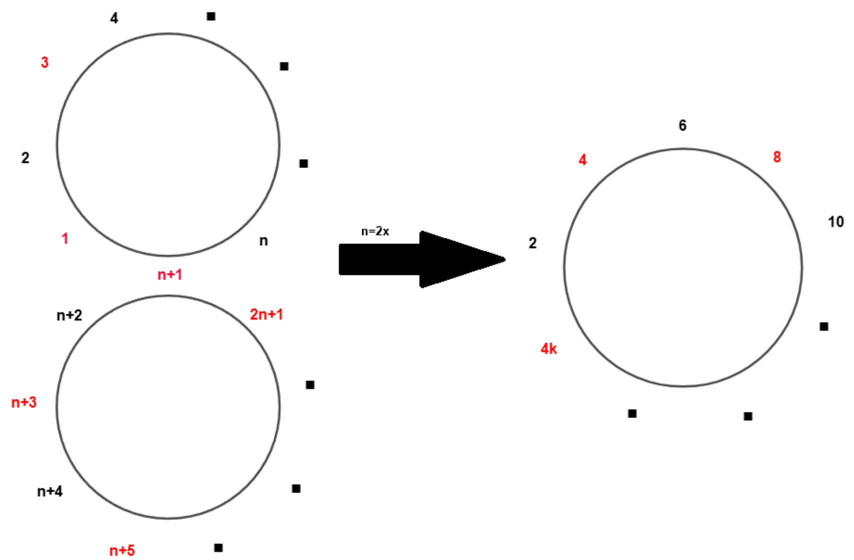

Figure 2: Visual representation of the $2n + 1 = 4x + 1$ case

3 Conclusion

In conclusion, by trying to discover the intricacies of the game 'Last one standing', we have used various mathematical methods from different fields, such as algorithm analysis, combinatorics, and number theory, which helped us finding the solutions to all the proposed generalizations. More specifically, we have used binary representations and recurrence relations for the basic game. Overall, our research demonstrates how structure,logic, and computation can be used to transform an apparently simple game into a complex mathematical puzzle.