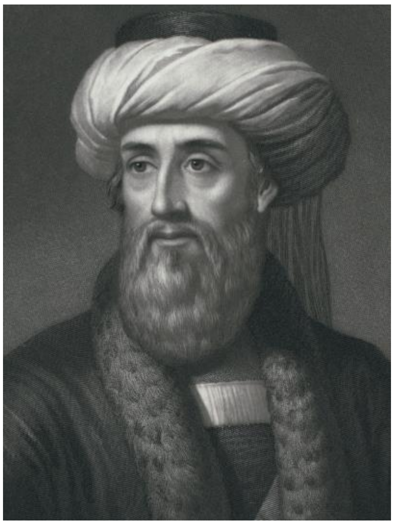
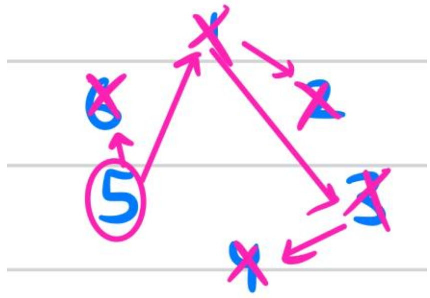
Surviving the Circle: The math of the Josephus Problem

In 67 A.D, during the Jewish Roman War, Joseph Flavius and 40 other soldiers were trapped in a cave and were being closed in on by the Roman army. The Jewish army (that Josephus was with) preferred to kill each other rather than be captured. As a result, they all formed a circle and planned for the first person to kill the second, the third to kill the fourth, the fifth to kill the sixth and continue until there is a sole survivor that will then kill themselves. Josephus much rather preferred to be captured than to commit suicide but he was worried if he told the others then they would turn on him and kill him. This problem then arose of what the winning seat would be so that Josephus could be in that position and give in to capture rather than killing himself.



So, to derive a pattern for the winning seat, W(n), in a group of n soldiers we should start off by looking at a group of 5 people. Number 1 kills number 2, number 3 kills number 4, number 5 kills number 1 and number 3 kills number 5, leaving 3 as the winner.

In a group of 7 people, 1 kills 2, 3 kills 4, 5 kills 6, 1 kills 3 and 5 kills 1. Leaving 1 as the winner.



Given that we use similar workings for numbers of n before and after such as 1,2,3,4,7 and 8. We can create a table with 2 headings of n and W(n) to try to help visualise if any patterns occur.

n	W(n)
1	1
2	1
3	3
4	1
5	3
6	5
7	7
8	1

In this table we can notice a few things. Such as, the number 1 being repeated many times and the fact that there are no even numbers. To make this easier to visualise, I have highlighted in yellow the numbers, n, where the winning seat is 1 and in blue the winning seat that is odd.

n	W(n)
1	1
2	1
3	3
4	1
5	3
6	5
7	7
8	1

Firstly, let's look at the numbers highlighted in yellow. The values of W(n) are 1 when the values of n are 1,2,4,8. These numbers look familiar to a family of numbers that we have been exposed to for a long time: the family of the powers of 2s. 1 representing 2^0, 2 representing 2^1, 4 representing 2^2 and 8 representing 2^3. So, we can now infer that for every number, n, that is a power of 2 then the corresponding W(n) will be the first seat.

Now, we can look at the numbers highlighted in yellow. Firstly, the number 3 appears twice, and they are all odd numbers. In the sequence of W(n) from n being 5 to n being 7, W(n) goes 3, 5, 7 all three of which are odd numbers but increase in intervals of 2 before being reverted to 1 as n becomes a power of 2.

In general, any number, N, can be written as a power of 2, 2^a, with some given residue which we will call l for now.

$N = 2^a + l$

For example, 65 can be written as $2^6 + 1$, where 1 represents the "l". 176 can be written as 128 + 48, which can be simplified to 128 + 32 + 16 which is just $2^7 + 2^5 + 2^4$ where $2^5 + 2^4$ represent the something.

This nicely links to the topic of binary which can also be represented by the Josephus problem. In binary, numbers are just written as the powers of 2. So, to take 176, this was written with the biggest power of 2 being 7. This means in binary form it would need 7 numbers, representing every power of 2 until 7. And the power of 2 that it is composed of will be written as 1 and the powers of 2 it is not composed of can be written as 0.

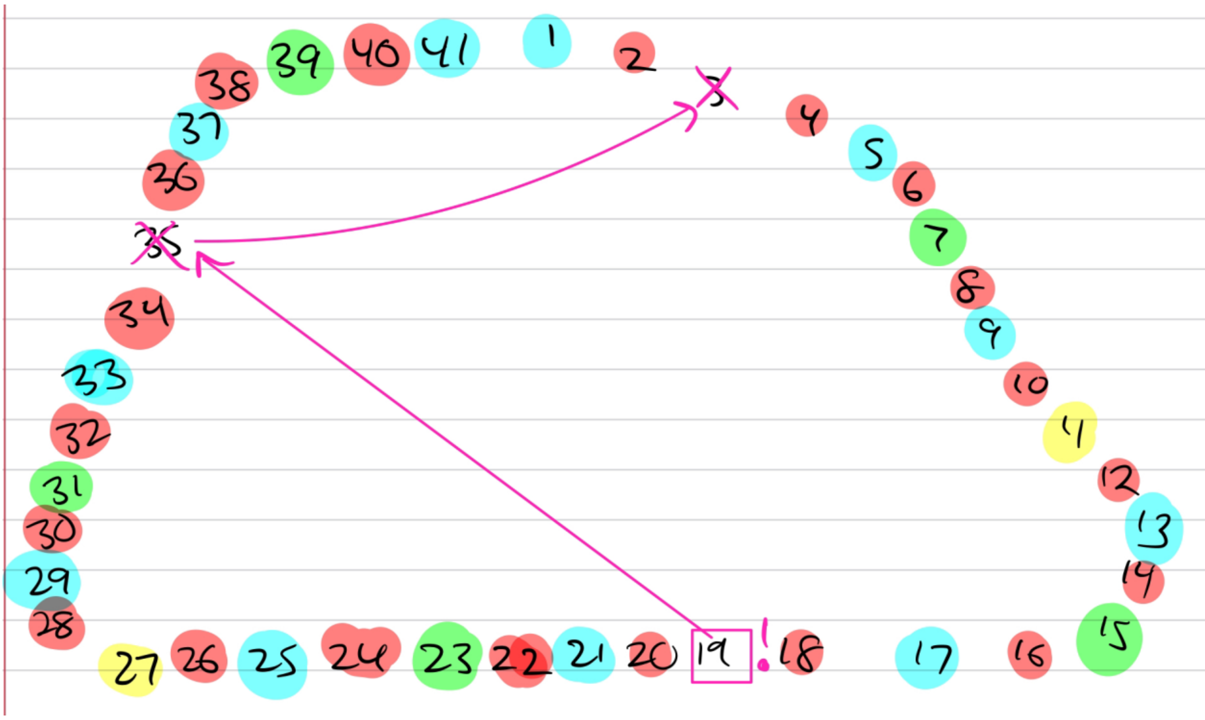
$176 = 2^7 + 2^5 + 2^4$

In binary:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	1	0	0	0	0

Now not in terms of binary, our form is $2^a + l$, so in the Josephus problem we can deduce that after l steps whoever's turn it is to kill someone will be the winner because remaining will just be a power of 2. Therefore, the winning seat can just be written as $2l + 1$. With a slight caveat being that n MUST be greater than 2^a and l must be less than 2^a.

So, in Josephus's scenario they were 41 people. 41 can be written as 32 + 9 which in the form $2^a + l$ is $2^5 + 9$. 9 multiplied by 2 being 18. 18 add 1 being 19. Therefore, in theory the winning seat should be 19. To prove that this is correct I can draw a sketch to show it.



With the red representing the first iteration, blue representing the second iteration, green representing the third and yellow representing the fourth iteration.

In binary code, we can even find a quicker solution to find W(n) just using n written in binary. So, to take 41 for an example. 41 can be written as 32 + 8 + 1 which is $2^5 + 2^3 + 2^0$. In binary, this is written as 101001. If you take the first number, 1 and move it all the way to the end you get 010011 which is $2^4 + 2^1 + 2^0$ which is 16 + 2 + 1 which is 19. We have already proved that W(n) is 19 therefore this is a further valid method.

This problem first appeared in mathematical literature in the 17th century but over time the Josephus problem found its way into algorithmic thinking because of its modular structure and recursive nature. By nature, we know the problem is recursive because to solve for n you must solve for n-1 and divide and conquer is needed in this problem which is an algorithmic design paradigm. In modular arithmetic it is a way of writing numbers with a remainder which we call mod. This is an important computer science topic for things such as cryptography, hashing and cyclic buffers.

In the real world, it could be used in operating systems to schedule tasks, circular data structures and load balancing. Furthermore, in other subjects like economics this could be used with game theory and its simulation because it is an elimination-style game where players are eliminated based on their position.

Perhaps some applications of this to things that may have not been created could be in musical chairs where people are removed from a circle, and you must try to find a sequence where there is a seat you can sit in to always win, this could be simulated just like the game theory simulation with the use of AI.

This could also be used in an island survival game where players are starving, and they must choose a person to kill- slightly gory- for food and to survive longer and you must find the player number that survives the longest.

In conclusion, this early algorithm problem is an interesting one that explores the idea of elimination in cycles which is why modern-day computer science loves it and has been integrated into it and can be explored creatively.