Mathematics in OI (II)

(HKOI 2015/16 Training Advanced Group Session 6)

12 MARCH 2016 (AM SESSION)

Jeffrey Hui Pak Nam

Mathematics in OI (I) – Intermediate (April 9)

CONTENTS:

Greatest Common Divisor (G.C.D) / Euclidean Algorithm

Least Common Multiple (L.C.M.)

Factorial Powers

Modular Arithmetic / Modular Inverse / **Chinese Remainder Theorem**

Finding Primes

High Precision Arithmetic (H.P.A.)

Partial Sum / Difference Operation

Mathematics in OI (II)

Euler's Phi Function

Recurrence Relation

The Josephus Problem

Combination & Permutation
⟶ Twelvefold Way *(We will skip this part today.)*
⟶ **Lucas' Theorem**

Catalan Number
⟶ The Drunken Man Problem

Euler's Phi Function (Introduction)

Euler's Phi Function φ (called *Phi*)

For every positive integer n,

$\varphi(n)$ = the number of positive integers not greater than n and coprime to n.

(Coprime means that $gcd(n, m) = 1$.)

For example: $\varphi(5) = 4$ (Corresponding m: $1, 2, 3, 4$.)

Without using your computer, find $\varphi(1)$ and $\varphi(12)$.

Beginners warm-up exercise 1.1: *input* n; *output* $\varphi(n)$. $(1 \leq n \leq 10\ 000)$

Euler's Phi Function (Value of φ)

$\varphi(n) =$ the number of positive integers not greater than n and co-prime to n.

$$= n \prod_{p \mid n} \left(1 - \frac{1}{p}\right)$$

Here, p is a prime.

For example, when $n = 12 = 2^2 \times 3$, chosen p: $2, 3$.

(Although 6 divides 12, $p = 6$ is not considered because 6 is composite!)

$\varphi(12) = 12 \times \left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{3}\right) = 12 \times \frac{1}{2} \times \frac{2}{3} = 4.$ (m: $1, 5, 7, 11$)

Beginners warm-up exercise 1.2: *input* n; *output* $\varphi(n)$. $(1 \leq n \leq 10^{12})$

Euler's Phi Function (Properties of $\varphi(p^k)$)

For any prime p, $\varphi(p) = p - 1$.

(Reason: for $m = 1 \ldots (p-1)$, $gcd(p, m) = 1$ while $gcd(p, p) = p > 1$.)

For any prime p, $\varphi(p^k) = (p-1)p^{k-1}$. ($k$ is any positive integer.)

(Reason: The only prime divisor of p^k is p. Only multiples of p have common divisors greater than 1 with p^k: they are p, $2p$, $3p$, …, p^k (p^{k-1} in total).)

Euler's Phi Function (Multiplicative Property)

For any positive integers m, n, if $gcd(m, n) = 1$,

$\varphi(mn) = \varphi(m)\varphi(n)$.

(For MO students, think about *Chinese Remainder Theorem*!)

A Pair of Residue modulo-and-coprime to m & Residue modulo-and-coprime to n will generate exactly ONE Residue modulo-and-coprime to mn.

Different pairs will generate different Residue modulo-and-coprime to mn.

If p is a prime divisor of mn, then p must be a prime divisor of m or n.

Euler's Phi Function (Independent Residues)

If the last slide is difficult to understand, here is an example to illustrate.

Take $m = 5, n = 6$ as an example!

$\varphi(5) = 4$ (coprime numbers: $1, 2, 3, 4$)

$\varphi(6) = 2$ (coprime numbers: $1, 5$)

$\varphi(30) = 8$ (coprime numbers: $1, 7, 11, 13, 17, 19, 23, 29$)

mod 5 (% 5)	mod 6 (% 6)	mod 30 (%30)
1	1	1
2	1	7
3	1	13
4	1	19

mod 5 (% 5)	mod 6 (% 6)	mod 30 (%30)
1	5	11
2	5	17
3	5	23
4	5	29

Euler's Phi Function (Proof: Value of φ)

Based on what we have learnt just now, we have:

$\varphi(p^k) = (p-1)p^{k-1} = p^k\left(1-\frac{1}{p}\right)$ (for any prime p)

$\varphi(mn) = \varphi(m)\varphi(n)$ (for any positive coprime integers m, n)

If we write n by prime factorization: ${p_1}^{k_1}{p_2}^{k_2}\dots{p_r}^{k_r}$, then

$$\varphi(n) = \varphi\left({p_1}^{k_1}{p_2}^{k_2}\dots{p_r}^{k_r}\right) = \prod_{i=1}^{r}\varphi\left({p_i}^{k_i}\right) = \prod_{i=1}^{r}{p_i}^{k_i}\left(1-\frac{1}{p_i}\right) = n\prod_{i=1}^{r}\left(1-\frac{1}{p_i}\right)$$

Euler's Phi Function (Euler's Theorem)

For any positive coprime integers a and n,

$a^{\varphi(n)} \equiv 1 \pmod{n}$

Theorem with Special Case: <u>Fermat's Little Theorem</u>

For any positive integer a and prime p, if a is not divisible by p,

$a^{p-1} \equiv 1 \pmod{p}$

Euler's Phi Function (Misconception of Euler's Theorem)

For any positive coprime integers a and n,

$a^{\varphi(n)} \equiv 1 \pmod{n}$

Notice that: $\varphi(n)$ is NOT necessary the smallest power of a congruent to 1 modulo n. Can you give me a counterexample?

$\varphi(7) = 6$ & $gcd(2,7) = 1$; but $2^3 = 8 \equiv 1 \pmod{7}$

Therefore, *(Exercise 1.3)* if you are asking to find the smallest positive integer k such that it satisfies $a^k \equiv 1 \pmod{n}$, <u>NEVER</u> answer $\varphi(n)$ immediately!

Euler's Phi Function (Applications)

For any positive coprime integers a and n,

$a^{\varphi(n)} \equiv 1 \pmod{n}$

Modular Inverse of a:

Since $a \cdot a^{\varphi(n)-1} \equiv 1 \pmod{n}$, $a^{-1} \equiv a^{\varphi(n)-1} \pmod{n}$

Further application: RSA cryptosystem *(See Wikipedia / Read PowerPoint of **2015** Feb 14 PM Session: Miscellaneous CS Topics (III))*

Recurrence Relation (Introduction)

If the <u>initial</u> terms (one or more) are given, each further term of the sequence is defined as a <u>function</u> of the preceding terms.

Typical Example: Fibonacci Sequence

$F_0 = 0, F_1 = 1$; for any positive integer $n > 1, F_n = F_{n-1} + F_{n-2}$.

In the following slides, we will use the following recurrence relation as an example: (you will know why later! ☺)

$C_1 = 0, C_2 = 1$; for any positive integer $n > 2, C_n = C_{n-1} + 2C_{n-2}$.

Beginners Warm-up Exercise 2.1: *input* n, *output* C_n. $(1 \leq n \leq 60)$

Recurrence Relation (Example Program 1)

```
C[1] = 0, C[2] = 1;
for (int i = 3; i <= n; i++)
      C[i] = C[i - 1] + 2 * C[i - 2];
return C[n];
```

Comments:

Time Complexity: $O(n)$

Memory Complexity: $O(n)$

// You need some improvements! 😐

Recurrence Relation (Example Program 2)

```
a = 0, b = 1;
if (n == 1) return a;
for (int i = 3; i <= n; i++){
        temp = b;
        b = 2 * a + b;
        a = temp;
} return b;
```

Comments:

Time Complexity: $O(n)$

Memory Complexity: $O(1)$ ☺

Recurrence Relation ($O(n)$ Time Complexity)

In the last 2 programs, Time Complexity is $O(n)$.

$C_1 = 0$, $C_2 = 1$; for any positive integer $n > 2$, $C_n = C_{n-1} + 2C_{n-2}$.

Beginners Warm-up Exercise 2.1: *input* n, *output* C_n. $(1 \leq n \leq 60)$

This n is very small!!! ☺☺☺

Exercise 2.2: *Input* positive integers n and m, *output* $C_n \pmod{m}$.
$(1 \leq n \leq 10^{18}, 1 \leq m \leq 10^9)$

Recurrence Relation (Mathematical Approach)

Let's see how a mathematics student finds the general term of the recurrence relation without using computer.

Generating Function:

$$G(x) = C_1 + C_2 x + C_3 x^2 + \cdots$$

$$C_n = \frac{1}{(n-1)!} \cdot \frac{d^{n-1} G(x)}{dx^{n-1}} \Big|_{x=0}$$

Recurrence Relation (Finding General Term)

$$G(x) = C_1 + C_2x + C_3x^2 + C_4x^3 + \cdots$$

$$xG(x) = C_1x + C_2x^2 + C_3x^3 + \cdots$$

$$2x^2G(x) = 2C_1x^2 + 2C_2x^3 + \cdots$$

$$(2x^2 + x - 1)G(x) = (C_1 - C_2)x - C_1 = (0 - 1)x - 0 = -x$$

$$G(x) = \frac{x}{(1-2x)(1+x)} \equiv \frac{A}{1-2x} + \frac{B}{1+x} \qquad \rightarrow \quad x \equiv A(1+x) + B(1-2x)$$

Sub. $x = 0.5$ and -1 respectively, we obtain $A = \frac{1}{3}$ and $B = -\frac{1}{3}$ respectively.

Since $\frac{1}{1-rx} = 1 + rx + r^2x^2 + \cdots$, we have $G(x) = \frac{1}{3}\{(1 + 2x + 2^2x^2 + \cdots) - [1 + (-1)x + (-1)^2x^2 + \cdots]\}$

$$G(x) = \sum_{n=0}^{\infty} C_{n+1}x^n = \frac{1}{3}\sum_{n=0}^{\infty}[2^n - (-1)^n]x^n$$

Comparing the coefficients, we have $C_{n+1} = \frac{1}{3}[2^n - (-1)^n]$, i.e. $C_n = \frac{1}{6} \times 2^n + \frac{1}{3} \times (-1)^n$.

Recurrence Relation (Characteristic Equations)

$C_1 = 0$, $C_2 = 1$; for any positive integer $n > 2$, $C_n = C_{n-1} + 2C_{n-2}$.

$C_n = C_{n-1} + 2C_{n-2}$
$x^2 = x + 2$, i.e. $x^2 - x - 2 = (x-2)(x+1) = 0$.

If the characteristic equation has two distinct roots x_1 and x_2, then
$C_n = A{x_1}^n + B{x_2}^n$, where A and B are constants.
(Determined by the values of initial terms C_1 and C_2.)

You can get the same result by this method!

Recurrence Relation (Fibonacci Sequence)

$F_0 = 0, F_1 = 1$; for any positive integer $n > 1$, $F_n = F_{n-1} + F_{n-2}$.

$F_n = F_{n-1} + F_{n-2}$: Characteristic Equation is: $x^2 - x - 1 = 0$.

$x = \frac{-(-1)\pm\sqrt{(-1)^2-4(1)(-1)}}{2(1)} = \frac{1\pm\sqrt{5}}{2}$.

$F_n = A{x_1}^n + B{x_2}^n$, where A and B are constants.

In fact, $F_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right]$.

Recurrence Relation (Floating Point Problem)

Exercise 2.2: *input* positive integers n and m, *output* $C_n \pmod m$.

$(1 \le n \le 10^{18}, 1 \le m \le 10^9)$

You may find general term yourself $C_n = \frac{1}{6} \times 2^n + \frac{1}{3} \times (-1)^n$ and just output the answer by using divide-and-conquer.

However, if we are asking this problem:

Exercise 2.2: *input* integers n and m, *output* $F_n \pmod m$.

$(0 \le n \le 10^{18}, 1 \le m \le 10^9)$

$F_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right]$ (You need to face huge floating point error! ☹)

Recurrence Relation (Matrix?)

What HKEAA requests in HKDSE. (Module 2, Year 2014)

12. (b) A sequence is defined by

$$x_1 = 0 \ , \quad x_2 = 1 \quad \text{and} \quad x_n = x_{n-1} + 2x_{n-2} \quad \text{for} \quad n = 3, 4, 5, \ldots \ .$$

It is known that this sequence can be expressed in the matrix form $\begin{pmatrix} x_n \\ x_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_{n-1} \\ x_{n-2} \end{pmatrix}$.

Using the result of (a)(iii), express x_n in terms of n .

(3 marks)

HKDSE M2 2014 Q.12(b)… MATRIX?!

Recurrence Relation (Matrix Representation)

$C_1 = 0, C_2 = 1$; for any positive integer $n > 2$, $C_n = C_{n-1} + 2C_{n-2}$. Thus,

$$\begin{pmatrix} C_n \\ C_{n-1} \end{pmatrix} = \begin{pmatrix} C_{n-1} + 2C_{n-2} \\ C_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} C_{n-1} \\ C_{n-2} \end{pmatrix}$$

$F_0 = 0, F_1 = 1$; for any positive integer $n > 1$, $F_n = F_{n-1} + F_{n-2}$. Thus,

$$\begin{pmatrix} F_n \\ F_{n-1} \end{pmatrix} = \begin{pmatrix} F_{n-1} + F_{n-2} \\ F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_{n-1} \\ F_{n-2} \end{pmatrix}$$

Recurrence Relation (Matrix Representation)

$$\begin{pmatrix} F_{n+1} \\ F_n \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_n \\ F_{n-1} \end{pmatrix}$$

$$\begin{pmatrix} F_{n+1} \\ F_n \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_{n-1} \\ F_{n-2} \end{pmatrix}$$

$$\cdots$$

$$\begin{pmatrix} F_{n+1} \\ F_n \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n \begin{pmatrix} F_1 \\ F_0 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Recurrence Relation (Matrix Exponential)

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^2 = \begin{pmatrix} 1 \cdot 1 + 1 \cdot 1 & 1 \cdot 1 + 1 \cdot 0 \\ 1 \cdot 1 + 0 \cdot 1 & 1 \cdot 1 + 0 \cdot 0 \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^4 = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}^2 = \begin{pmatrix} 2 \cdot 2 + 1 \cdot 1 & 1 \cdot 1 + 1 \cdot 0 \\ 1 \cdot 2 + 1 \cdot 1 & 1 \cdot 1 + 1 \cdot 0 \end{pmatrix} = \begin{pmatrix} 5 & 3 \\ 3 & 2 \end{pmatrix}$$

$$\begin{pmatrix} F_{n+1} \\ F_n \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \rightarrow \quad \begin{pmatrix} F_5 \\ F_4 \end{pmatrix} = \begin{pmatrix} 5 & 3 \\ 3 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 5 \cdot 1 + 3 \cdot 0 \\ 3 \cdot 1 + 2 \cdot 0 \end{pmatrix} = \begin{pmatrix} 5 \\ 3 \end{pmatrix}$$

Hence, $F_5 = 5$ and $F_4 = 3$. (Check yourself!)

Time Complexity: $O(\log_2 n)$ ☺☺☺

The Josephus Problem (Introduction)

N people $\{1, 2, 3, \cdots, N\}$ standing in a circle are waiting to be killed. ☹

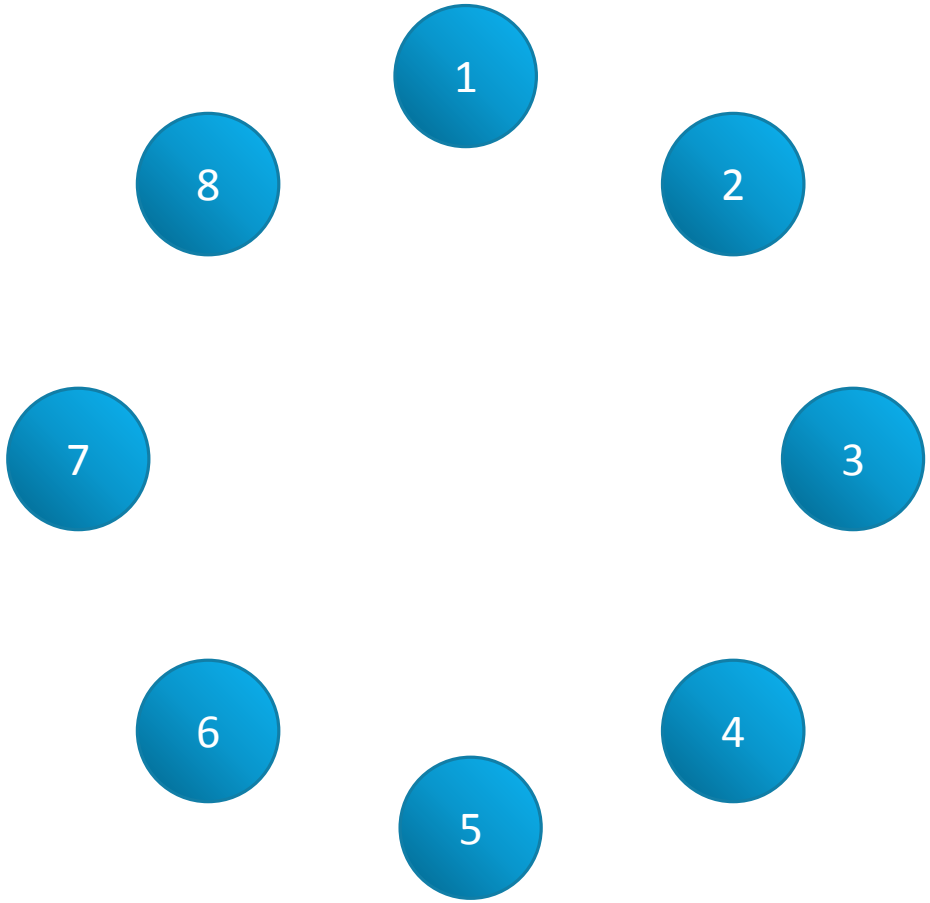
Kill the Kth person, skip the next $K - 1$ people.

Loop until one survivor remained.

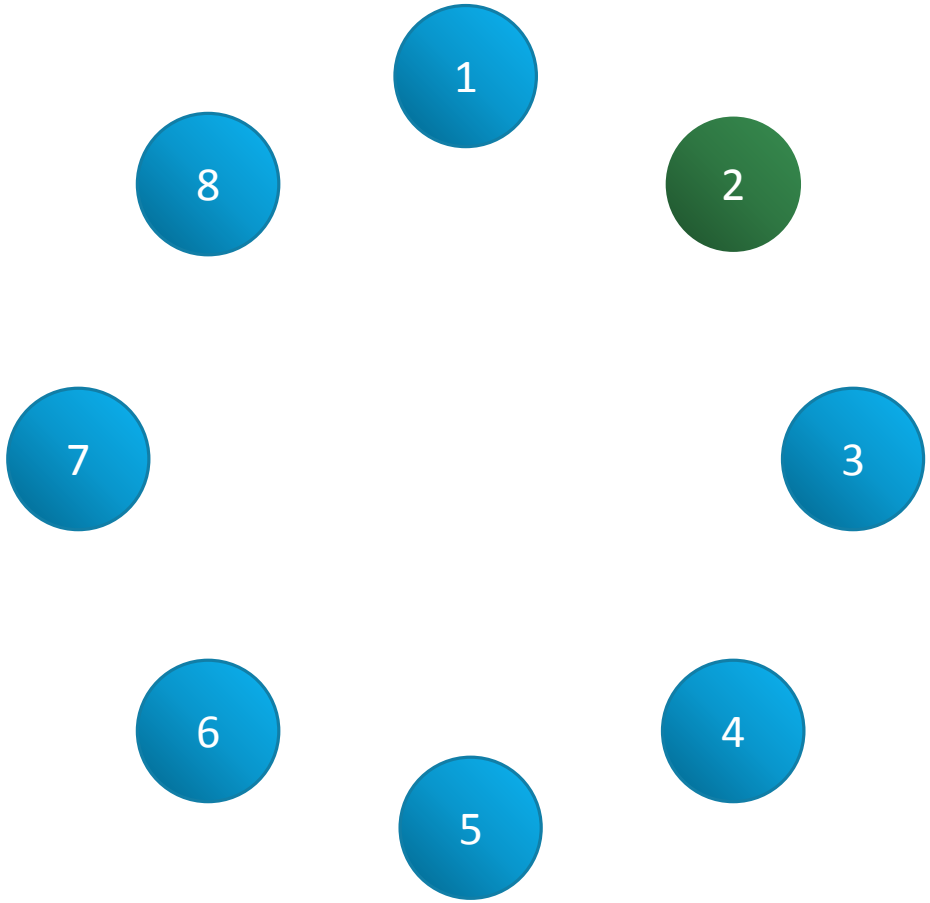
In the following slides, we will take $N = 8$ as an example!

Simplified Version: $K = 2$.

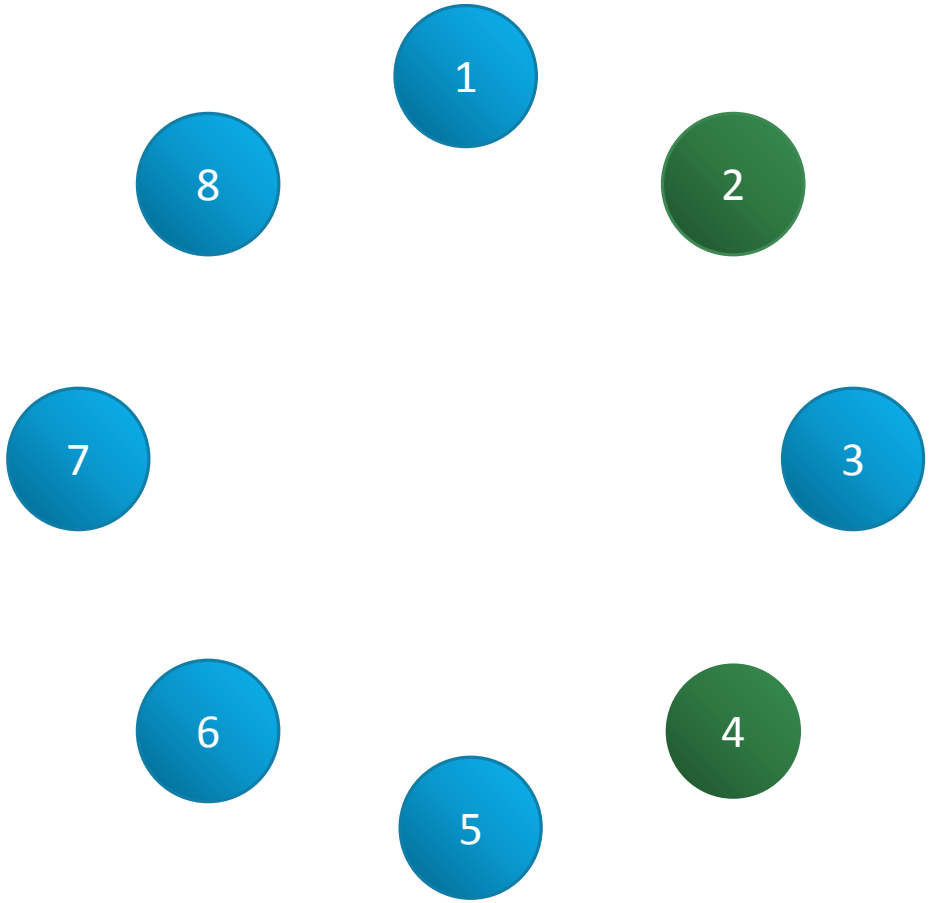
The Josephus Problem (Illustration)



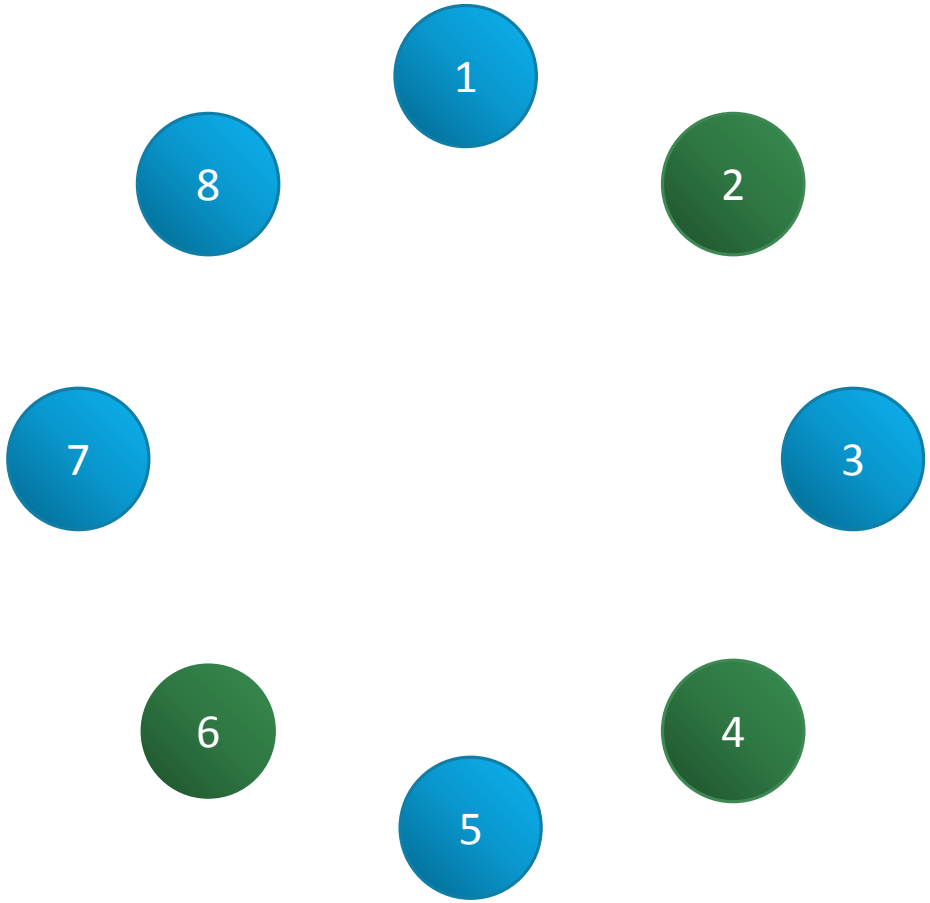
The Josephus Problem (Illustration)



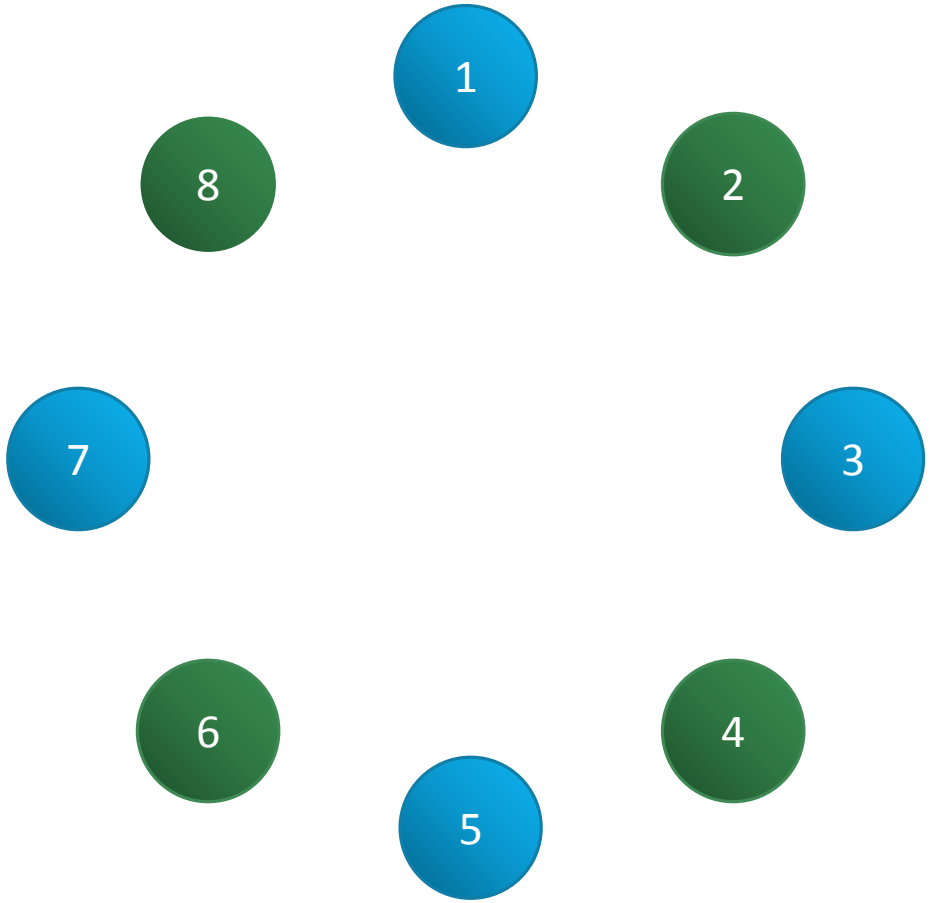
The Josephus Problem (Illustration)



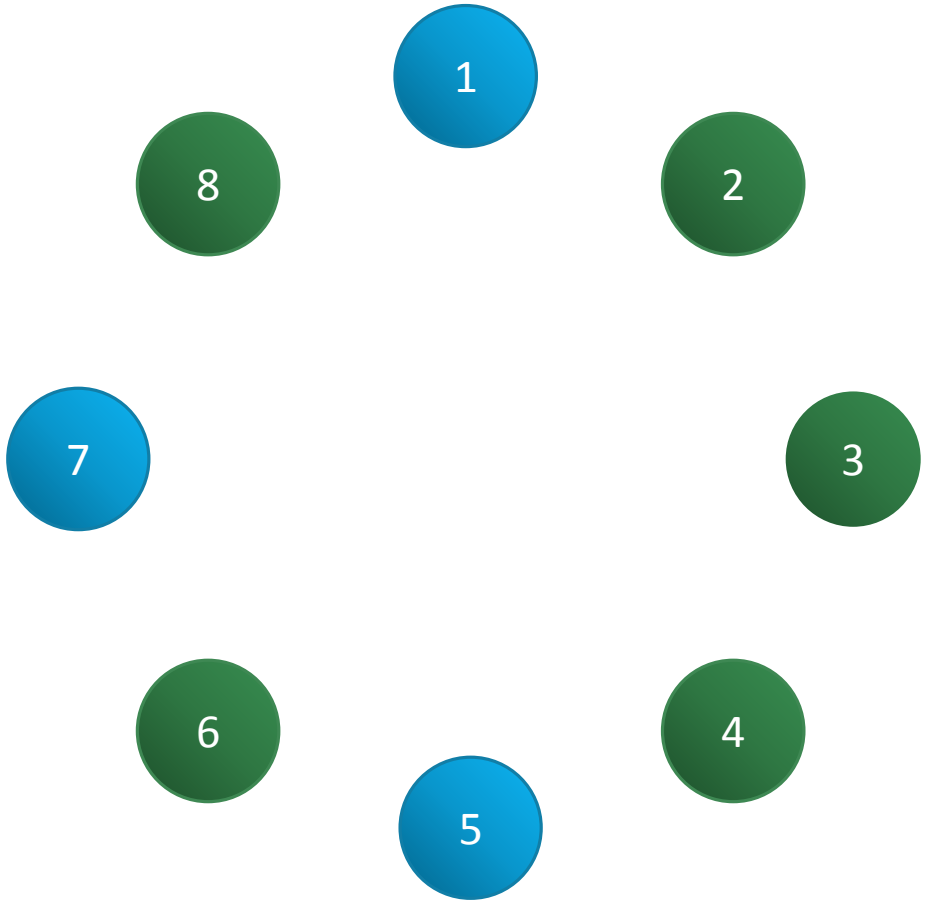
The Josephus Problem (Illustration)



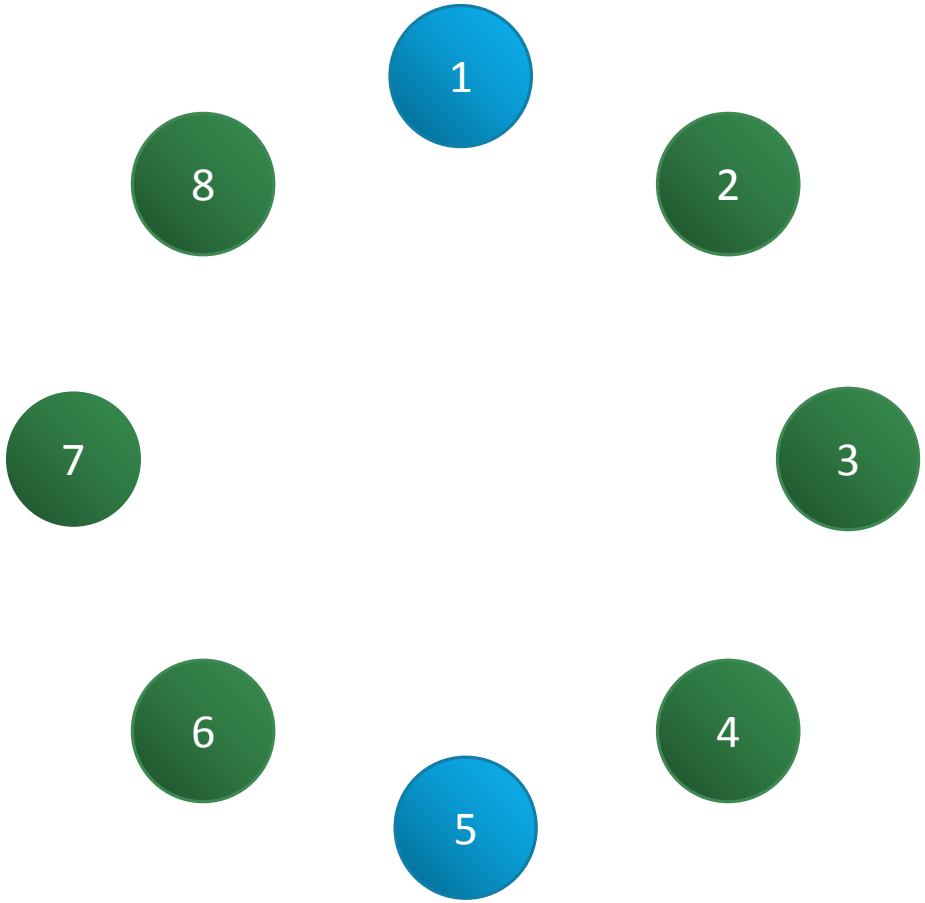
The Josephus Problem (Illustration)



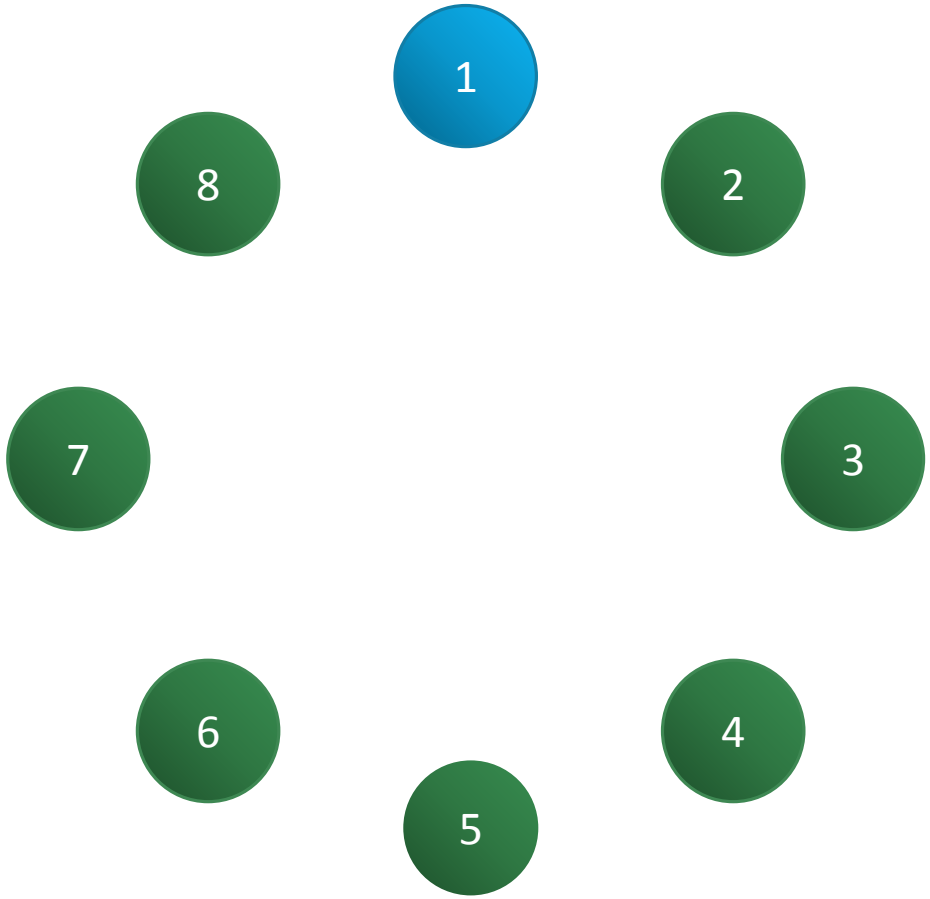
The Josephus Problem (Illustration)



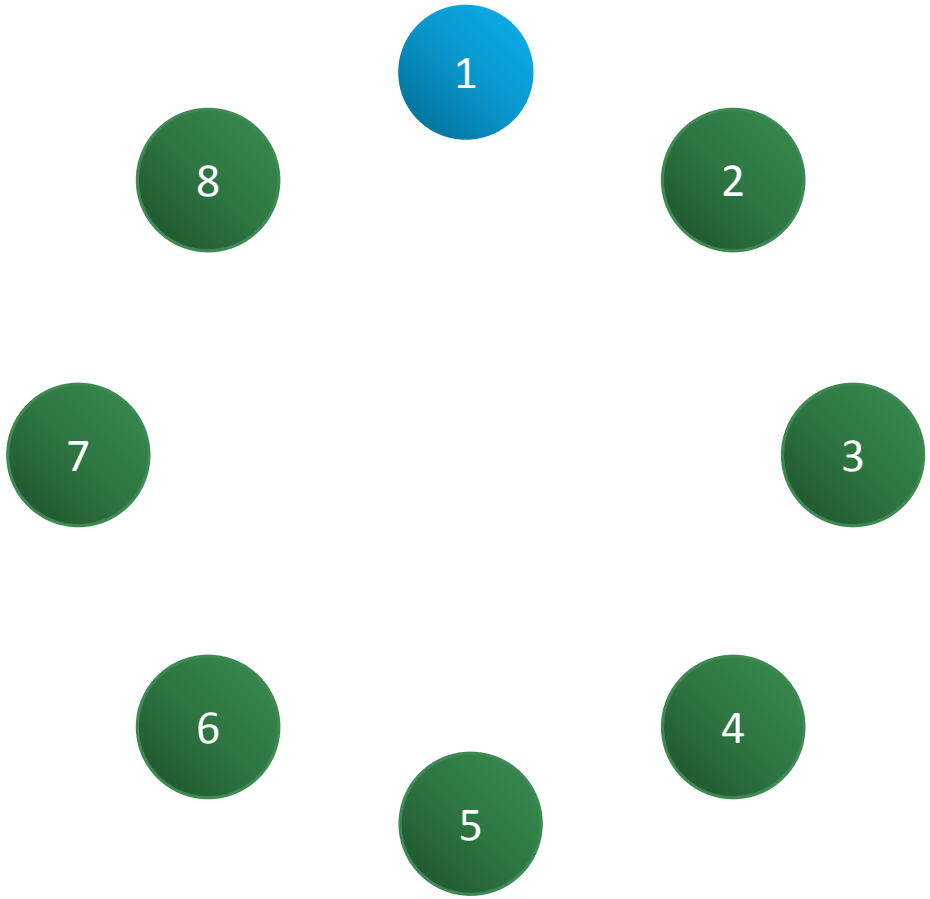
The Josephus Problem (Illustration)



The Josephus Problem (Illustration)



The Josephus Problem (Illustration)



The Josephus Problem (Observation)

We may simulate the game until only one is remained!

Time Complexity?

$O(NK)$ (☹ for large N and K)

Always check who is alive and who is dead every round, too time consuming…

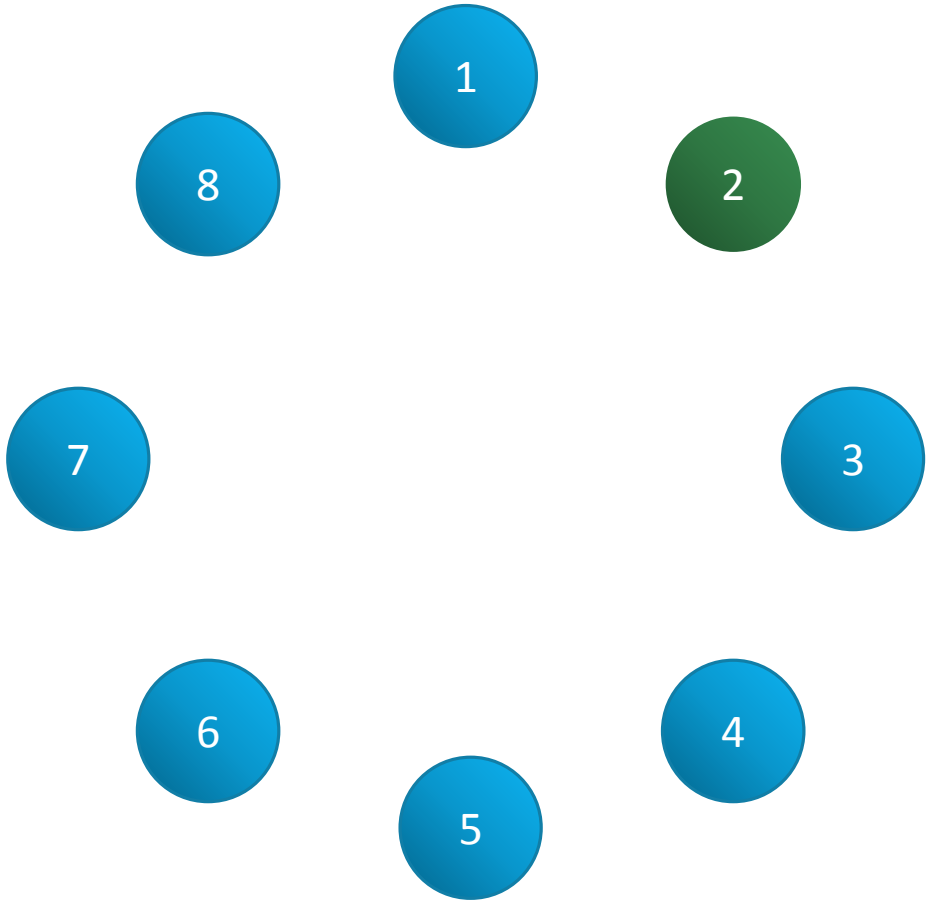
The Josephus Problem (Observation)

Make every round look similar to the 1st round. (Why?)

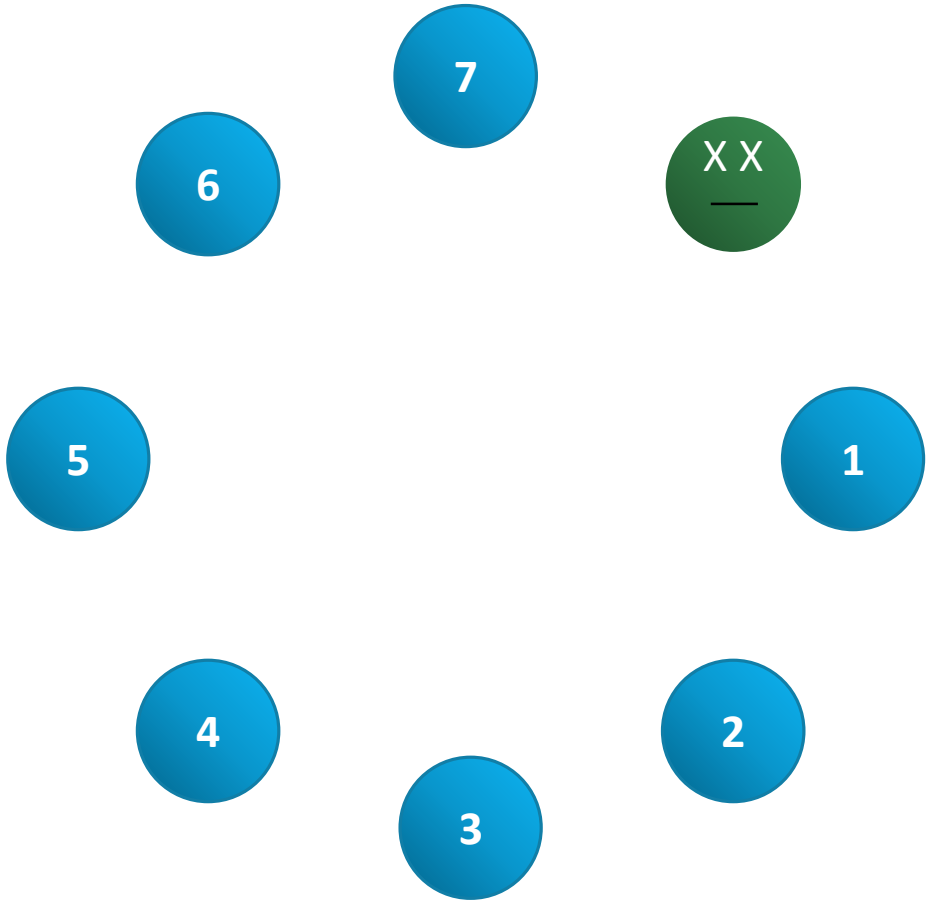
Because we always know that the person with label K (Or for large $K > N$, the person with label $(K-1)\ mod\ N + 1$ will be killed next!

Why don't we give a new labeling for the next round?

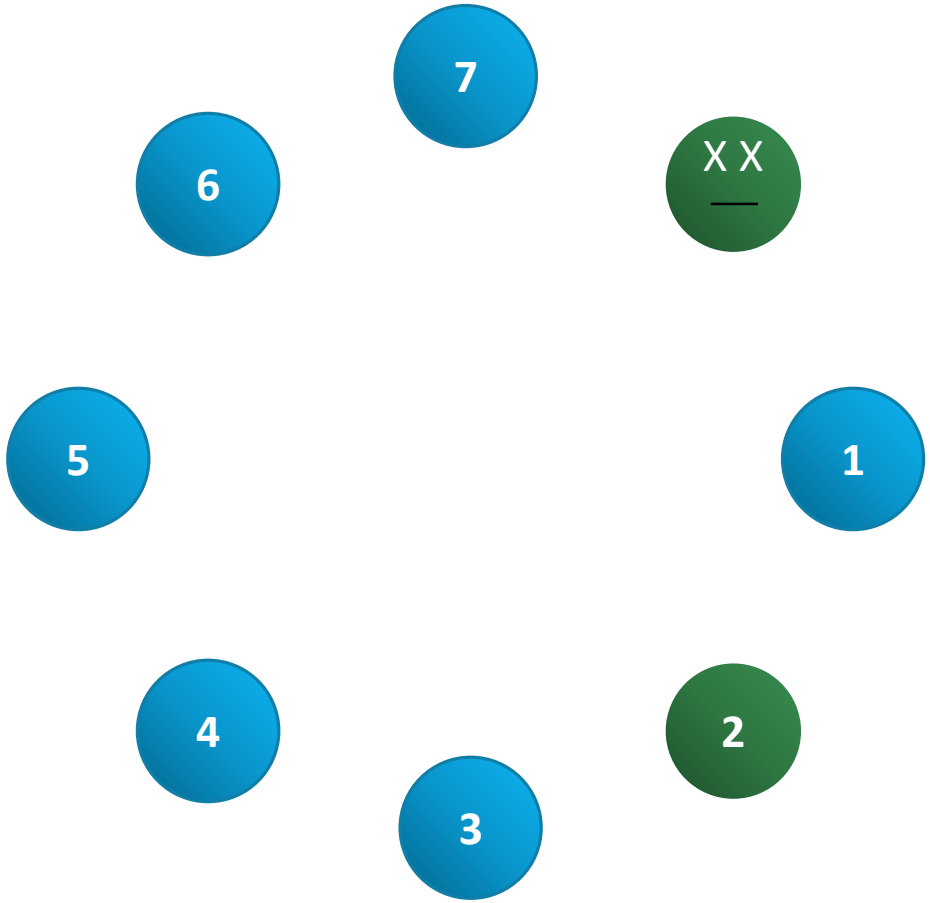
The Josephus Problem (Illustration)



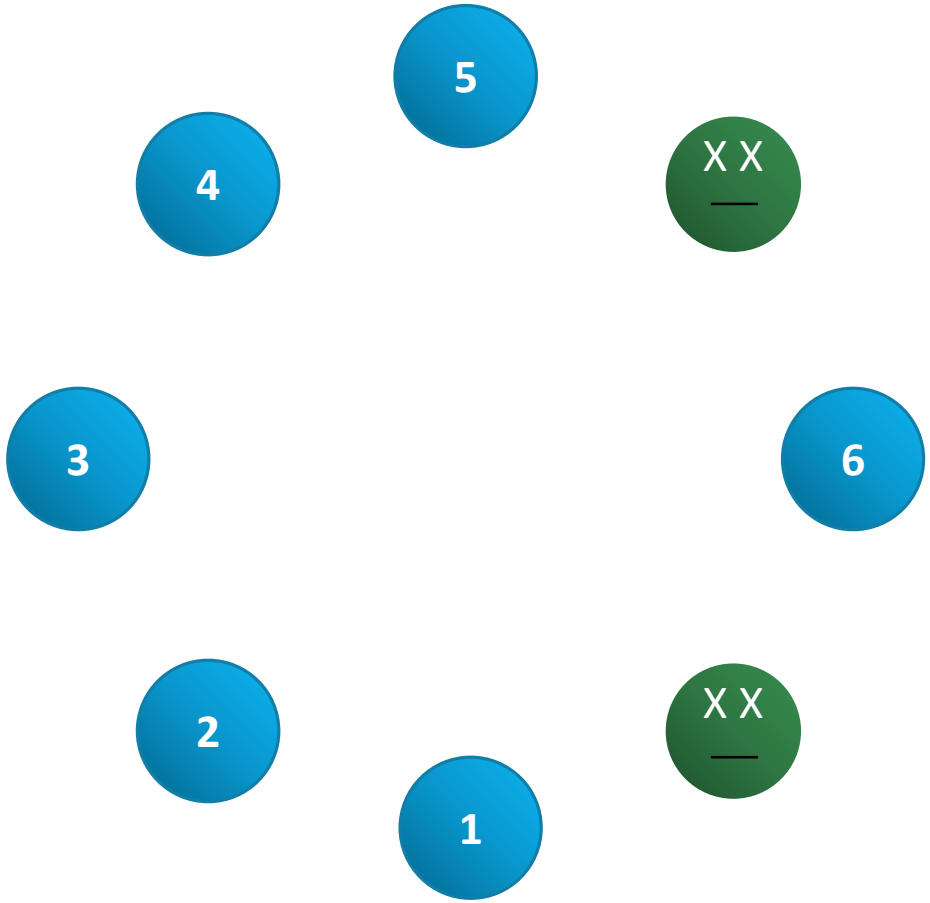
The Josephus Problem (Illustration)



The Josephus Problem (Illustration)



The Josephus Problem (Illustration)



The Josephus Problem (Table-form)

Original (M ppl)	1	2	3	4	5	6	7	8
New ($M-1$ ppl)	7	DEAD	1	2	3	4	5	6

Original (M ppl)	1	2	3	4	5	6	7
New ($M-1$ ppl)	6	DEAD	1	2	3	4	5

What is our task? Find the initial label of the survivor?

What is the label of the survivor in the last round?

If there are M survivors,
we call $f(M)$ be the label of "last survivor" when there are M survivors.

Can you write $f(M)$ in terms of $f(M-1)$?

The Josephus Problem (Table-form)

Original (M ppl)	1	2	3	4	5	6	7	8
New ($M-1$ ppl)	7	DEAD	1	2	3	4	5	6

We may observe that for some columns, original – new = 2

$f(M) = f(M-1) + 2?$

1st column… ☹

$1 \equiv 9 \ (mod\ 8)$ → Use mod?

$f(M) = (f(M-1) + 1)\ mod\ M + 1?$

The Josephus Problem (Results)

When $K = 2$, we have the formula $f(M) = (f(M-1)+1)\ mod\ M + 1$?

Now, how about $K > 2$?

Use the same observations, you may also write an expression of $f(M)$ in terms of $f(M-1)$ and K.

Try to understand the previous slides. Try to get the answer! ☺

Twelvefold Way (Introduction)

We are now going to put 4 balls into 3 boxes. How many ways are there?

Do all of you get the same answer?

Twelvefold Way (Considerations?)

Balls: Identical? Or Distinct?

Boxes: Identical? Or Distinct?

Allow more than one ball into one box?

Allow empty boxes?

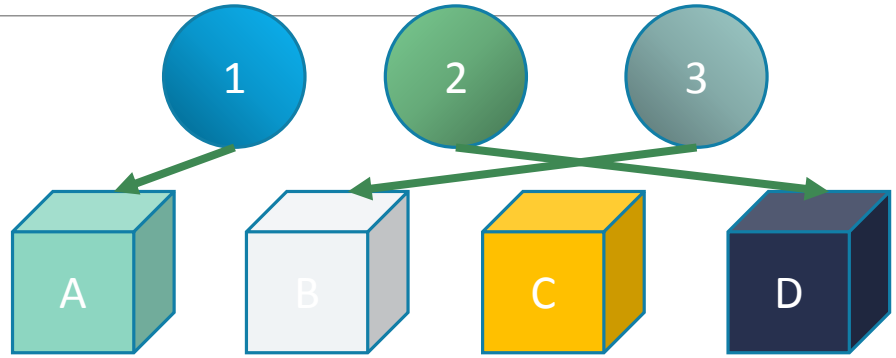
Twelvefold Way (Concept of Function)

$f: X \to Y$ (A function from set X to set Y)

If set X consists of N balls and set Y consists of K boxes, function f from X to Y is to link *(put)* the balls to the boxes.

Twelvefold Way (Injective Function)

$f: X \to Y$ (A function from set X to set Y)



Injective function: preserves distinctness.

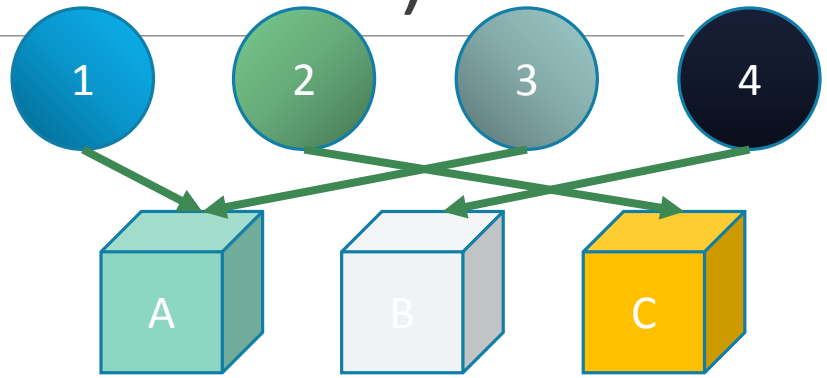
Never maps distinct elements of X to the same element of Y.

What does it mean in the case of X: balls and Y: boxes?

(Answer: Never maps more than one ball into the same box.)

Twelvefold Way (Surjective Function)

$f: X \to Y$ (A function from set X to set Y)



Surjective function:

Every element of Y has at least one corresponding element of X.

What does it mean in the case of X: balls and Y: boxes?

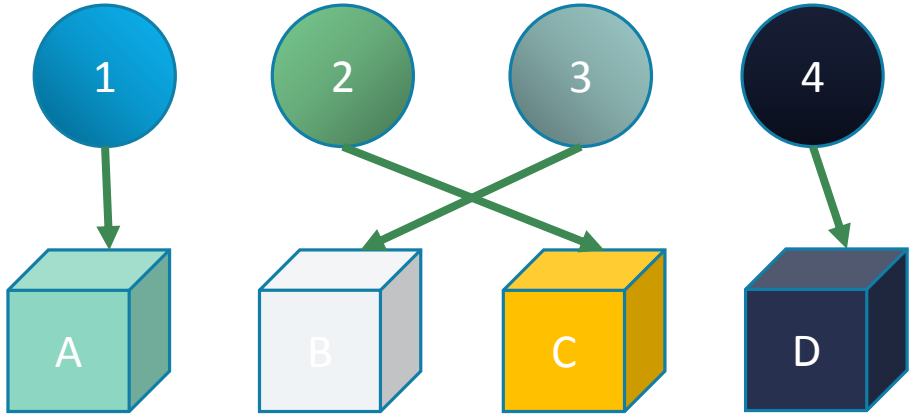
(Answer: For every box, it contains at least one ball. (i.e. No empty box!))

Twelvefold Way (Bijective Function)

For the injective case:

What is the constraints between number of balls and number of boxes such that at least one possible mapping exist?

How about surjective?



Bijective function is both Injective & Surjective.

Why is bijective case not considered as one of the Twelvefold Way?

Twelvefold Way (Results)

We are now going to put N balls into M boxes. How many ways are there?

N Balls	X Boxes	Any f	Injective f	Surjective f
Labeled	Labeled	M^N	P_N^M	$M! \cdot S(N, M)$
Unlabeled	Labeled	C_N^{M+N-1}	C_N^M	C_{M-1}^{N-1}
Labeled	Unlabeled	$\sum_{i=1}^{m} S(N, i)$	$\begin{cases} 1, n \le m \\ 0, n > m \end{cases}$	$S(N, M)$
Unlabeled	Unlabeled	$\sum_{i=1}^{m} p(N, i)$	$\begin{cases} 1, n \le m \\ 0, n > m \end{cases}$	$p(N, M)$

$p(N, M) =$ the number of partitions of the number N into M parts.

$S(N, M) =$ the number of ways to partition a set of N numbers into M non-empty sets.

Search *"Partition (Number Theory)"* & *"Stirling numbers of the second kind"* for more details.

Twelvefold Way (Permutations)

Given n distinct balls, pick r balls one by one without replacement. How many ways are there?

(Answer: P_r^n)

$$P_r^n = \frac{n!}{(n-r)!} = n(n-1)(n-2)\cdots(n-r+2)(n-r+1)$$

Ball-and-Box Problem:

We are now going to put r <u>labeled</u> balls into n <u>labeled</u> boxes, given that each box contains <u>at most one</u> ball. (The number of ways = C_r^n)

Twelvefold Way (Combinations)

Given n distinct balls, randomly select r balls. How many selections are there?

(Answer: C_r^n)

$$C_r^n = \frac{n!}{(n-r)!r!} = \frac{n(n-1)(n-2)\cdots(n-r+2)(n-r+1)}{1\times2\times3\times\cdots\times(r-1)\times r}$$

Ball-and-Box Problem:

We are now going to put r <u>identical</u> balls into n <u>labeled</u> boxes, given that each box contains <u>at most one</u> ball. (The number of ways = C_r^n)

Permutations & Combinations

Twelvefold way is not our focus today, because you need to know how to compute P_r^n and C_r^n correctly by program.

Exercise 4.1:

Input positive integers n and r where $n \geq r$, *output* C_r^n.

(It is guaranteed that the output is within 32-bit. ☺)

Permutation (Calculation of P_r^n)

$$P_r^n = \frac{n!}{(n-r)!} = n(n-1)(n-2)\cdots(n-r+2)(n-r+1)$$

Easy to calculate! ☺

```
int product = 1;
for (int i = n; i > n - r; i--)
     product *= i;
return product;
```

Combination (Calculation of C_r^n)

$$C_r^n = \frac{n!}{(n-r)!r!} = \frac{n(n-1)(n-2)\cdots(n-r+2)(n-r+1)}{1\times2\times3\times\cdots\times(r-1)\times r}$$

```
int product = 1;
for (int i = n; i > n - r; i--)
      product *= i;
for (int i = 1; i <= r; i++)
      product /= i;
return product;
```

C_{15}^{30} lies within 32-bit integer range. But can this program generate this number?

Combination (Pascal Triangle)

You may have learnt Pascal Triangle in your secondary school.

$$
\begin{array}{ccccccccccc}
 & & & & & 1 & & & & & \\
 & & & & 1 & & 1 & & & & \\
 & & & 1 & & 2 & & 1 & & & \\
 & & 1 & & 3 & & 3 & & 1 & & \\
 & 1 & & 4 & & 6 & & 4 & & 1 & \\
1 & & 5 & & 10 & & 10 & & 5 & & 1
\end{array}
$$

$C_0^n = C_n^n = 1; C_r^n = C_{r-1}^n + C_{r-1}^{n-1}.$

Combination (Dynamic Programming: 2D)

You may refer to the PowerPoint of JAN 30 AM Session (Adv.): *Dynamic Programming (I)*.

Given a $N \times M$ grid, find out the number of paths from top left cell to bottom right cell. **<u>Restriction:</u>** You can move down or right only, but no obstacles! ☺

Take $N = 5$ and $M = 4$ as an example.

1	1	1	1	1
1	2	3	4	5
1	3	6	10	15
1	4	10	20	35

Combination (Dynamic Programming: 2D)

1	1	1	1	1
1	2	3	4	5
1	3	6	10	15
1	4	10	20	**35**

C_0^0	C_0^1	C_0^2	C_0^3	C_0^4
C_1^1	C_1^2	C_1^3	C_1^4	C_1^5
C_2^2	C_2^3	C_2^4	C_2^5	C_2^6
C_3^3	C_3^4	C_3^5	C_3^6	$\boldsymbol{C_3^7}$

You can only move down 3 cells and move right by 4 cells (in any order). (Example: RDDRRDR) *D = Down; R = Right*

7-letter string. How many ways to place three letter-D?

$35 = C_3^7$

Combination (Dynamic Programming: 2D)

We may use addition method to overcome overflow problem.

So, *Exercise 4.1* is done!

But, is it a good method?

Memory Complexity: $O(N)$

Time Complexity: $O(NM)$ ☹

Combination (Observation on C_r^n)

$$C_r^n = \binom{n}{r} \times \frac{(n-1)(n-2)\cdots(n-r+2)(n-r+1)}{1\times2\times3\times\cdots\times(r-1)} = \binom{n}{r} \times C_{r-1}^{n-1}$$

We can not guarantee that n is divisible by r, but C_{r-1}^{n-1} must be an integer!

```
int nCr (int n, int r){
      if (r == 0) return 1;
      return nCr(n - 1, r - 1) * n / r;
}
```

Memory Complexity: $O(1)$; Time Complexity: $O(r)$ ☺☺☺

Combination (Observation on C_r^n)

$$C_r^n = \binom{n}{r} \times \frac{(n-1)(n-2)\cdots(n-r+2)(n-r+1)}{1\times2\times3\times\cdots\times(r-1)} = \binom{n}{r} \times C_{r-1}^{n-1}$$

Can you use a non-recursive approach to compute C_r^n? Yes, but be careful!

```
int nCr = 1;
for (int i = 1; i <= r; i++){
        nCr *= n - r + i;
        nCr /= i;
}
```

Be careful the division sign, you cannot get a perfect score in *Exercise 4.1* because you used `int` to store nCr.

Combination (Upgraded Exercise II)

In *Exercise 4.1*, C_r^n is within 32-bit, so n and r can be very small.

Exercise 4.2:

Input positive integers n, r and a prime p where $n \geq r$, *output* $C_r^n \pmod p$.

Lucas' Theorem

For non-negative integers n and r and a prime p,

We may represent n and r in the form of summation of power of p.

$n = n_k p^k + n_{k-1} p^{k-1} + \cdots + n_2 p^2 + n_1 p + n_0$

$r = r_k p^k + r_{k-1} p^{k-1} + \cdots + r_2 p^2 + r_1 p + r_0$

$$C_r^n \equiv \prod_{i=0}^{k} C_{r_i}^{n_i} \ (mod\ p)$$

If $n < r$, $C_r^n = 0$. You may search the proof of Lucas' Theorem from the Internet.

Lucas' Theorem (Example)

If $n = 2016$, $r = 312$ and $p = 17$,

We may represent n and r in the form of summation of power of 17.

$n = 6 \times 17^2 + 16 \times 17 + 10$

$r = 1 \times 17^2 + 1 \times 1 + 6$

$$C_{312}^{2016} \equiv C_1^6 \times C_1^{16} \times C_6^{10} \equiv 6 \times 16 \times 210 \equiv 20160 \equiv 15 \ (mod\ 17)$$

Check the above results from *Wolfram Alpha*! <u>Type</u>: C(2016, 312) mod 17

Lucas' Theorem

If p is very large, which maybe larger than r, then calculating $C_{r_i}^{n_i}$ is difficult! ☹

$C_r^n \cdot (r!) = n(n-1)(n-2)\cdots(n-r+2)(n-r+1)$

$C_r^n \equiv n(n-1)(n-2)\cdots(n-r+2)(n-r+1) \cdot (r!)^{-1} \pmod{p}$

By Fermat's Little Theorem, $(r!)^{p-1} \equiv (r!)(r!)^{p-2} \equiv 1 \pmod{p}$

$(r!)^{-1} \equiv (r!)^{p-2} \pmod{p}$

Division ☹ → Multiplication ☺

Combination (Upgraded Exercise III)

In *Exercise 4.2*, we take C_r^n modulo p where p is a prime.

Exercise 4.3:

Input positive integers n, r and m where $n \geq r$, *output* $C_r^n \pmod{m}$.

(It is guaranteed that m is square-free, i.e. it is not divisible by any perfect squares greater than 1.)

Combination (Example)

If $n = 2016, r = 312$ and $m = 221 = 17 \times 13$,

In *Exercise 4.2*, we have already computed C_{312}^{2016} modulo 17 and modulo 13 respectively. We have $C_{312}^{2016} \equiv 15\ (mod\ 17)$ and $C_{312}^{2016} \equiv 2\ (mod\ 13)$.

(Chinese Remainder Theorem) We can find C_{312}^{2016} modulo 221.

How? Apply Extended Euclidean Algorithm. **(Answer: 15)**

Read the PowerPoint of Mathematics in OI (I) on April 9. ☺

Check the above results from *Wolfram Alpha*! Type: C(2016, 312) mod 221

Combination (Upgraded Exercise IV)

In *Exercise 4.3*, we take C_r^n modulo m where m is a square-free.

Can you guess what I am going to ask next?

Exercise 4.4:

Input positive integers n, r, a and prime p where $n \geq r$, *output* $C_r^n \pmod{p^a}$.

(It is guaranteed that p^a is within 32-bit.)

I will show you some ideas/approaches because it is quite difficult.

NOTE: This approach shown in the next two slides may not be the best solution.

Combination (Example)

If $n = 14, r = 8, p = 7$ and $a = 2$,

<u>Step 1:</u> Count the number of maximal power of p, say it is b.

How? Read the PowerPoint of <u>Mathematics in OI (I)</u> on April 9.

$$b = \sum_{k=1}^{\infty} \left\lfloor \frac{2016}{11^k} \right\rfloor - \left\lfloor \frac{312}{11^k} \right\rfloor - \left\lfloor \frac{2016 - 312}{11^k} \right\rfloor = 1$$

<u>Step 2:</u> Rewrite C_r^n.

We have $C_r^n \equiv p^b \times \boldsymbol{n}! \times (\boldsymbol{r}!)^{-1} \times \left((\boldsymbol{n} - \boldsymbol{r})!\right)^{-1} \ (mod\ p^a)$

Bold $\boldsymbol{k}!$ means all factors of $\boldsymbol{p}$ have been taken out.

Combination (Example)

<u>Step 3:</u> Take out all factors p in the product form of C_r^n.

Example: Consider $C_8^{14} = \frac{14!}{8! \times 6!}$.

We have calculated that its maximal power of 7 is 1.

So $C_8^{14} \equiv 7^1 \times \mathbf{14}! \times (\mathbf{8}!)^{-1} \times (\mathbf{6}!)^{-1} \ (mod\ 7^2)$

Then consider $\mathbf{14}! \times (\mathbf{8}!)^{-1} \times (\mathbf{6}!)^{-1} \ (mod\ 7^1)$

$$\mathbf{14}! = 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 1 \times 8 \times 9 \times 10 \times 11 \times 12 \times 13 \times 2$$
$$\equiv (6!)^2 \times 2 \equiv (-1)^2 \times 2 \equiv 2 \ (mod\ 7)$$ [Wilson's Theorem]

$\mathbf{8}! \equiv \mathbf{6}! \equiv 720 \equiv 6 \ (mod\ 7)$. Hence, $(\mathbf{8}!)^{-1} \equiv (\mathbf{6}!)^{-1} \equiv 6 \ (mod\ 7)$

Therefore, $C_8^{14} \equiv 7 \times 2 \times 6 \times 6 \equiv 14 \ (mod\ 7^2)$.

Combination (Conclusion)

If you can solve *Exercises 4.3* and *4.4*, you can solve the followings.

Exercise 4.5:

Input positive integers n, r and m, *output* $C_r^n \pmod m$.

A Drunken Man

A drunken man was standing beside a wall. On each second he moves left or right by 1 m at random.

What is the probability that he returns to the original position in k seconds without hitting the wall?

A Drunken Man

What is the probability that he returns to the original position in k seconds without hitting the wall?

Is it possible that k is odd?

Let $k = 2n$. Then n left and n right move.

Number of all possible outcomes $= 2^{2n}$.

We have to find the number of moving sequence such that the man returns to original position without hitting the wall.

A Drunken Man

If the wall is removed, the number of ways is C_n^{2n}.

But the wall is still here. Let A_n be the number of ways.

If the first time the man returns to his original position is at the $(2i)^{th}$ second

The first move must be rightward, and the $(2i)^{th}$ move must be leftward.

A Drunken Man (Observation)

From 2 to $2i - 1$ second,
due to the definition of $2i$, he cannot return to original position in this period.

For 2 & $2i - 1$ second,
he must be standing on 1 m on the right of original position.

Problem in this time interval reduced to a sub-problem with $(2i - 2)$ total moves.

Think of an invisible wall on his original position.

A Drunken Man (Observation)

A_{i-1} ways for the first $2i$ seconds

A_{n-i} ways for the remaining $2n-2i$ seconds
(The man may return to original position again.)

$$A_n = \sum_{i=1}^{n} A_{i-1} A_{n-i}$$

Calculate $A_1, A_2, A_3, \cdots, A_n$.

A Drunken Man (Solution)

Assume that the coordinate of the original position is $x = 0$.

The path is invalid when he reaches the coordinate $x = -1$ (because the man will hit the wall in this case.)

Denote L: Left; R: Right. Denote a path by a string with 'L' and 'R' with length $2n$.

Consider an invalid path, for example, **RRLRLLL**RLR. The man returns the original position after 10 seconds but he hits the wall because RRLRLLL, i.e. he reaches the coordinate $x = -1$ after 7 seconds.

A Drunken Man (Solution)

Example of invalid path: **RRLRLLL**RLR.

When the path of the drunken man is found to be invalid, we do not need to care the remaining steps because he has already hit the wall.

In the followings, we will flip the remaining steps when the coordinate $x = -1$ is reached.

That is, **RRLRLLL**<u>RLR</u> becomes **RRLRLLL**<u>LRL</u>.

According to the path **RRLRLLL**<u>LRL</u>, the drunken man will end at the coordinate $x = -2$.

A Drunken Man (Solution)

For any invalid path, we must flip the remaining steps when $x = -1$, such that the drunken man will end at $x = -2$.

On the other hand, any path ends at $x = -2$ must be invalid.

Therefore, the total number of invalid paths should be C_{n+1}^{2n}.

(Consider a string with length $2n$, which consists of $n + 1$ 'Left-steps'.)

A Drunken Man (Solution)

End at the original position: number of paths $= C_n^{2n}$.

End at $2\,m$ on the left of the original position: number of paths $= C_{n-1}^{2n}$.

The number of ways that the man never arrives at the considered position

$$= C_n^{2n} - C_{n-1}^{2n} = \frac{C_n^{2n}}{n+1}$$

This is called the Catalan Number.

Catalan Number (Applications)

The number of valid string consisting of n pairs of parenthesis.
()()(), (())(), ()(()), (()()), ((()))

Reason: Open parenthesis ≥ Close parenthesis

The number of monotonic paths along the edges of a grid with $n \times n$ square cells.

Reason: Move right ≥ Move left

Catalan Number (Applications)

The number of binary trees with n nodes

The number of ways that a convex polygon with $(n+2)$ sides can be cut into triangles by connecting vertices with straight lines.

Remember this formula:

$$A_n = \sum_{i=1}^{n} A_{i-1} A_{n-i}$$

Thank you!

Any Questions about today's training?

PM Session: Old Trainers Forum

Venue: Room 712, William M. W. Mong Engineering Building

Don't miss it! ☺