**🚀 1. Next index with wrap-around (circular array)**

csharp

CopyEdit

int next = (i + 1) % nums.Length;

✅ Example: if i = nums.Length - 1 (last index), this becomes 0.

**🔄 2. Previous index with wrap-around (circular array backward)**

csharp

CopyEdit

int prev = (i - 1 + nums.Length) % nums.Length;

✅ Ensures it stays non-negative. If i = 0, then it becomes nums.Length - 1.

**📝 LeetCode Math & Combinatorics Cheat Sheet**

**✅ Factorial**

n!=n×(n−1)×(n−2)×⋯×1n! = n \times (n-1) \times (n-2) \times \dots \times 1n!=n×(n−1)×(n−2)×⋯×1

**✅ Combinations (nCr)**

(nr)=n!r!(n−r)!\binom{n}{r} = \frac{n!}{r!(n-r)!}(rn​)=r!(n−r)!n!​

**Use:** To count the number of ways to choose r elements from n without caring about order.

**✅ Permutations (nPr)**

P(n,r)=n!(n−r)!P(n, r) = \frac{n!}{(n-r)!}P(n,r)=(n−r)!n!​

**Use:** To count the number of ways to arrange r elements out of n, where **order matters**.

**✅ Catalan Number (Parentheses, BSTs, Dyck paths)**

Cn=1n+1(2nn)=(2n)!(n+1)!⋅n!C\_n = \frac{1}{n+1} \binom{2n}{n} = \frac{(2n)!}{(n+1)! \cdot n!}Cn​=n+11​(n2n​)=(n+1)!⋅n!(2n)!​

**Use cases:**

* Number of valid parentheses strings with n pairs.
* Number of unique BSTs with n nodes.
* Number of ways to triangulate a polygon.

**✅ Power**

ab=a×a×a×…a^b = a \times a \times a \times \dotsab=a×a×a×…

* Use fast exponentiation (binary exponentiation) to compute a^b % m efficiently.

**✅ GCD (Greatest Common Divisor)**

gcd(a,b)\text{gcd}(a, b)gcd(a,b)

* Found using **Euclidean algorithm**.

**✅ LCM (Least Common Multiple)**

lcm(a,b)=a×bgcd⁡(a,b)\text{lcm}(a, b) = \frac{a \times b}{\gcd(a, b)}lcm(a,b)=gcd(a,b)a×b​

**✅ Sum of first n natural numbers**

1+2+3+⋯+n= n(n+1)2

1 + 2 + 3 + \dots + n = \frac{n(n+1)}{2}1+2+3+⋯+n=2n(n+1)​

**✅ Sum of first n squares**

12+22+32+⋯+n2=n(n+1)(2n+1)61^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}12+22+32+⋯+n2=6n(n+1)(2n+1)​

**✅ Sum of first n cubes**

13+23+33+⋯+n3=(n(n+1)2)21^3 + 2^3 + 3^3 + \dots + n^3 = \left( \frac{n(n+1)}{2} \right)^213+23+33+⋯+n3=(2n(n+1)​)2

**✅ Modular Arithmetic**

* **Add:** (a+b)%m(a+b) \% m(a+b)%m
* **Multiply:** (a×b)%m(a \times b) \% m(a×b)%m
* **Power:** Use binary exponentiation to compute (ab)%m(a^b) \% m(ab)%m.

**✅ Fibonacci numbers**

F0=0,F1=1,Fn=Fn−1+Fn−2F\_0 = 0, F\_1 = 1, F\_n = F\_{n-1}+F\_{n-2}F0​=0,F1​=1,Fn​=Fn−1​+Fn−2​

(Use matrix exponentiation or memoization for fast computation.)

**✅ Pascal’s Triangle for Combinations**

(nr)=(n−1r−1)+(n−1r)\binom{n}{r} = \binom{n-1}{r-1} + \binom{n-1}{r}(rn​)=(r−1n−1​)+(rn−1​)

**🚀 Quick Reference Table**

| **Formula** | **Use** |
| --- | --- |
| (nr)=n!r!(n−r)!\binom{n}{r} = \frac{n!}{r!(n-r)!}(rn​)=r!(n−r)!n!​ | Ways to choose r items from n |
| Cn=1n+1(2nn)C\_n = \frac{1}{n+1}\binom{2n}{n}Cn​=n+11​(n2n​) | Valid parentheses / BSTs / Dyck paths |
| P(n,r)=n!(n−r)!P(n,r) = \frac{n!}{(n-r)!}P(n,r)=(n−r)!n!​ | Ways to arrange r from n |
| gcd⁡(a,b)\gcd(a,b)gcd(a,b) | Greatest common divisor |
| lcm(a,b)=a×bgcd⁡(a,b)\text{lcm}(a,b)=\frac{a\times b}{\gcd(a,b)}lcm(a,b)=gcd(a,b)a×b​ | Least common multiple |
| ∑i=1ni=n(n+1)2\sum\_{i=1}^n i = \frac{n(n+1)}{2}∑i=1n​i=2n(n+1)​ | Sum of first n numbers |
| ∑i=1ni2=n(n+1)(2n+1)6\sum\_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}∑i=1n​i2=6n(n+1)(2n+1)​ | Sum of squares |
| ∑i=1ni3=(n(n+1)2)2\sum\_{i=1}^n i^3 = \left(\frac{n(n+1)}{2}\right)^2∑i=1n​i3=(2n(n+1)​)2 | Sum of cubes |