

Modular Arithmetic

For CP

Contents

- Modulo Operator (%)
- Modulo Addition
- Problem Set 1
- Modulo Multiplication
- Problem Set 2
- Modulo Division + Binary Exponentiation

The Modulo Operator

- For an integer a , and modulo (integer) m
- $b = a \% m$, b is the remainder a when divided by m
- i.e repeatedly subtracting m from a until $a < m$

Modular Addition

- Int a, int b, int mod
- Find $(a+b) \% \text{mod}$
- What if $(a+b)$ overflows?
- $(a+b) \% \text{mod} = ((a \% \text{mod}) + (b \% \text{mod})) \% \text{mod}$

Modular Multiplication

- Int a, int b, int mod
- Find $(a * b) \% \text{mod}$
- But $a * b$ may overflow
- $(a * b) \% \text{mod} = ((a \% \text{mod}) * (b \% \text{mod})) \% \text{mod}$

Problem #1

- <https://leetcode.com/problems/maximum-subarray-min-product/>
- <https://leetcode.com/problems/sum-of-floored-pairs/>
- <https://leetcode.com/problems/largest-palindrome-product/>

Binary Exponentiation

```
• public static long binpow(long a, long b, long m) {  
•   a %= m;  
•   long res = 1;  
•   while (b > 0) {  
•     if (b%2==1) {  
•       res = res * a % m;  
•     }  
•     a = a * a % m;  
•     b >>= 1;  
•   }  
•   return res;  
• }
```

Modular Division

- `public static long moddiv(long a, long b, long m) {`
- `return (a%m * binpow(b,m-2,m)%m)%m;`
- `}`

- <https://pi.math.cornell.edu/~morris/135/mod.pdf>
- https://en.wikipedia.org/wiki/Modular_multiplicative_inverse

Sieve of Eratosthenes

	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

Problem Set

- <https://codeforces.com/contest/1881/problem/D>
- (difficult) <https://codeforces.com/contest/1886/problem/D>