Today's content

- Recap
- Questions using modulo arithmetic
- optional content.

A % B = {Remainder when A is divided by B}

12 % 5 = 2

[Dividend = Div. * Quo. + Remainder]

A % B = Keep subtracting B from A until value < B.

30 % 7 = (30-7) : (23-7) : (16-7) : (9-7) : 2

40 % 7 = (40-7) : (33-7) : (26-7) : (19-7) : (12-7) : 5

// Why do we need mod? // to limit our range.

$-\infty$ to $+\infty$ } % 10 = [0 to 9]

a % M → {0 to M-1}

Modular Arithmetic

① $[(a+b)\ \%\ M = (a\%m + b\%m)\%M]$

$a=8 \quad b=5 \quad m=10$

$(a+b)\%m = (a\%m + b\%m)\%m$

③ $(8 + 5)\%10 = 3.$

② $(a+M)\ \%\ m = ((a\%m)+(M\%m))\%m = (a\%m)\%m$

$[(a+M)\%m = (a\%m)]$

③ $[(a*b)\ \%\ m = (a\%m * b\%m)\%m]$

④ $[(a-b)\ \%\ M = (a\%m - b\%m + m)\%m]$

$a=10, \ b=2, \ m=9$

$((10\%9) - (2\%9) + 9)\%9$

$= (1 - 2 + 9)\%9$

$= (8)\%9 = 8.$

$a=7, \ b=2, \ m=9$

$= (7-2)\%9 = 5\%9 = 5$

$(7 - 2 + 9)\%9$

$= (5+9)\%9 = 5\%9 = 5.$

④ $(a/b)\%m$ → After 2 questions.

Q: Given N +ve array elements. Calculate no. of pairs (i,j) such that `arr[i]+arr[j] % M = 0`.

Note → i != j and pair(i,j) is same as pair(j,i).

arr[6] : [4 7 6 5 8 3] , M = 3
(indices: 0 1 2 3 4 5)

pairs:

i	j	arr[i]+arr[j]			
0	3	4+5 = 9	⇒	9%3 = 0	ans = 5
0	4	4+8 = 12	⇒	12%3 = 0	
1	3	7+5 = 12	⇒	12%3 = 0	
1	4	7+8 = 15	⇒	15%3 = 0	
2	5	6+3 = 9	⇒	9%3 = 0	
2	3	6+5 = 11	⇒	11%3 = 2	✗

Idea-1. Consider all the pairs.

```
count = 0
for ( i = 0 ; i < N ; i++) {
    for (j = i+1 ; j < N ; j++) {
        if ((arr[i] + arr[j]) % m == 0) {
            count ++
        }
    }
}
return count
```

T.C → O(N²)
S.C → O(1)

Idea:- $(a+b)\%m$ $\rightarrow [0, m-1]$

$\Rightarrow [a\%m + b\%m]\%m = 0$

$(1, m-1)$: { Consider values of b such that $b\%m = m-1$ }

$(2, m-2)$: { consider values of b such that $b\%m = m-2$ }

$(3, m-3)$: { consider values of b such that $b\%m = m-3$ }

Generalization $(i, m-i)$

$(0, 0)$: Both values are same.

$(m\%2==0)$ $\left(\frac{m}{2}, m/2\right)$: Both values are same

$m/2$ case.

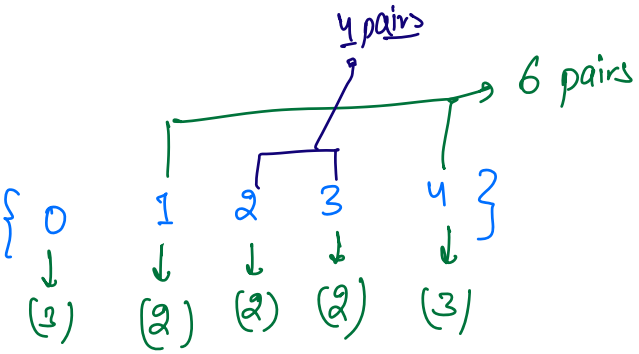
$m=6$. { 0 ✓ 1 2 3 4 5 }

Eg

arr[12] : [6 (0) 7 (1) 5 (2) 11 (3) 19 (4) 20 (5) 9 (6) 15 (7) 14 (8) 13 (9) 12 (10) 23 (11)], M=5

marr[12] : [1 (0) 2 (1) 0 (2) 1 (3) 4 (4) 0 (5) 4 (6) 0 (7) 4 (8) 3 (9) 2 (10) 3 (11)]

{2,5}
{2,7}
{5,7}



$^3C_2$ pairs.

$\Rightarrow \frac{3(3-1)}{2} = 3.$

[Total no. of pairs = 13]

$^NC_2 = \frac{N(N-1)}{2}$

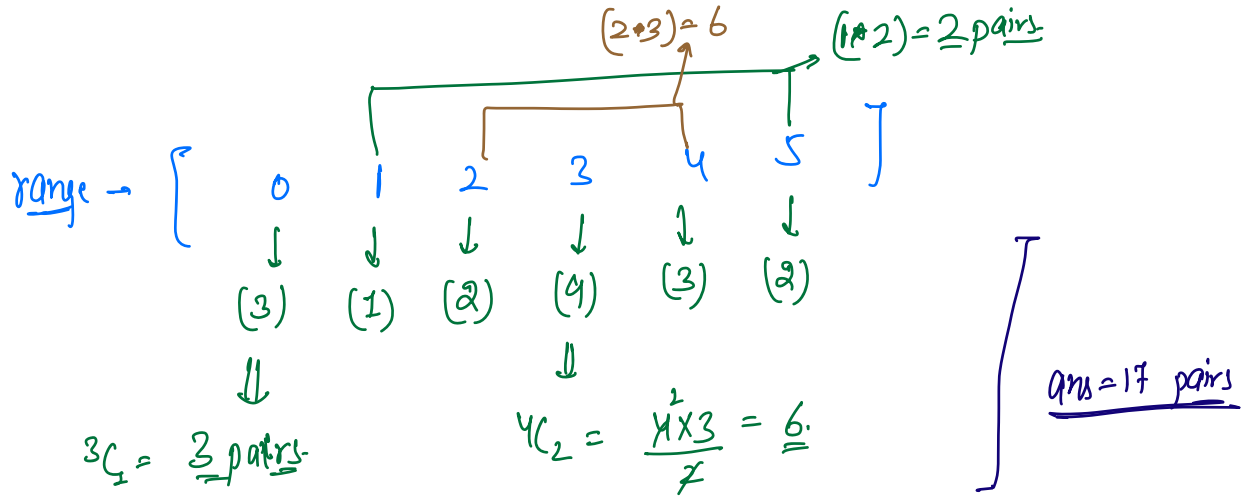
↳ In how many ways, you can select 2 elements from N elements.

arr[15] : [2 3 4 8 6 15 5 12 17 7 18 10 9 16 21] , m = 6

(0 1 2 3 4 5 6 7 8 9 10 11 12 13 14)

mod 7 - { 2 3 4 2 0 3 5 0 5 1 0 4 3 4 3 }

(0 1 2 3 4 5 6 7 8 9 10 11 12 13 14)



After taking mod → values will lie in range [0, m-1]

Find frequency of all these elements. [0, m-1].

① Array of size M

{todo}

{0, m-1}.

S.C = O(m)

② Hashmap.

{0 - m-1}

O(m), O(N)

{N > m, m > N} ⇓ O(N)

S.C = O(min(m, N))

{it is slightly better}

pseudo-code

```
HashMap < int, int > hm ;
// insert all the elements in hashmap after taking modulo
// with m.
for ( i = 0 ; i < N ; i++) {
        int val = arr[i] % m
        if ( hm.search(val) == true) { hm[val]++ }
        else { hm.insert (val, 1) }
}

ans = 0
x = hm[0] ;
ans += (x) (x-1) / 2 ;
if (m % 2 == 0) {
        x = hm[m/2] ;
        ans += (x) (x-1) / 2 ;
}
for ( i = 1 ; i < (m+1)/2 ; i++) {
    ans = ans + hm[i] * hm[m-i] ;
}
return ans
```

T.C → O(N+m)
S.C → O(min(N,m))

m=10 { 0 1 2 3 4 5 6 7 8 9 } i [1-4]

m=11 { 0 1 2 3 4 5 6 7 8 9 10 } [1-5]

{ Break till 10:37 }

Q) Given N distinct arr[] elements, where $0 \le arr[i] < N$

Replace arr[i] with arr[arr[i]]

S.C → O(1)

[All elements from 0 to N-1 are present.]

arr[5] = { 3 2 4 1 0 }

arr[] → [1 4 0 2 3]

arr[0] = arr[arr[0]] = arr[3] = 1

arr[1] = arr[arr[1]] = arr[2] = 4

arr[2] = arr[arr[2]] = arr[4] = 0

arr[3] = arr[arr[3]] = arr[1] = 2

arr[4] = arr[arr[4]] = arr[0] = 3

arr = { ~~3~~ 2 4 ~~1~~ 0 }

arr[4] = arr[0]

{ Swap:
You are losing old data }

arr[7]: [3 1 4 6 5 0 2]

[6 1 5 2 0 3 4]

arr[7]: [1 6 3 5 4 2 0]

[6 0 5 2 4 3 1]

Idea → → No swapping.

→ No extra space.

→ At arr[i], we need to store old data & new data.

// Big Bang :

		Day	Hour
		0	0
23	:	0	23
40	:	1	16
100	:	4	4
125	:	5	5
202	:	8	10
x	:	x/24	x%24

$\frac{x}{24}$ = Quotient = days

x%24 = Remainder = Hours.

x → /n → new | old

x → %n → old | new

#todo

x → /n → old

x → %n → new

⇒ $x = old * n + new$

/n = old

%n = new.

{arr[i] → old * N + new}

arr[7] : [3*7+6 (0) 1*7+1 (1) 4*7+5 (2) 6*7+2 (3) 5*7+0 (4) 0*7+3 (5) 2*7+4 (6)]

newvalues : 6 1 5 2 0 3 4

	idx.	value.	
arr[0] = arr[arr[0]]	arr[0] / 7 = 3	arr[3] / 7 = 6	, add 6 to arr[0]
arr[1] = arr[arr[1]]	arr[1] / 7 = 1	arr[1] / 7 = 1	, add 1 to arr[1]
arr[2] = arr[arr[2]]	arr[2] / 7 = 4	arr[4] / 7 = 5	, add 5 to arr[2]
arr[3] = arr[arr[3]]	arr[3] / 7 = 6	arr[6] / 7 = 2	, add 2 to arr[3]
arr[4] = arr[arr[4]]	arr[4] / 7 = 5	arr[5] / 7 = 0	add 0 to arr[4]
arr[5] = arr[arr[5]]	arr[5] / 7 = 0	arr[0] / 7 = 3	add 3 to arr[5]
arr[6] = arr[arr[6]]	arr[6] / 7 = 2	arr[2] / 7 = 4	add 4 at arr[6].

pseudo-code:

```
for(i=0; i<N; i++) {
    arr[i] = arr[i]*N
}
```

```
for(i=0; i<N; i++) {
    // arr[i] = arr[arr[i]]
    idx = arr[i]/N
    val = arr[idx]/N
    arr[i] += val
}
```

```
for(i=0; i<N; i++) {
    arr[i] = arr[i]%N
}
```

arr[i] = old*N + new

%N

arr[i] = new.

T.C → O(N)
S.C → O(1)

[* → Storing two values at one index]

$(a/b) \% M = (a\%m \;/\; b\%m) \% m$

$(a/b) \% m = (a * b^{-1}) \% m = (a\%m * b^{-1}\%m) \% m.$

$b^{-1}\%m$ ⇓ Inverse modulo.

[Given a, m, $a^{-1}\%m$ exists only if $gcd(a,m) = 1$

↳ proof. {heavy mathematics}

↳ Greatest common divisor.]

$gcd(a,m) = 1$

Given a, m, $\{b = a^{-1} \% m\} \rightarrow [1, m-1]$

$(a*b) \% m = 1$

// What should be the value of b such that $(a*b) \% m = 1$.

$b \rightarrow [1, m-1]$.

$a = 7$, $m = 10$, $gcd(a,m) = 1$, $\{b = a^{-1} \% m = 3.\}$

$b = 1 \Rightarrow (7*1) \% 10 \neq 1$

$b = 2 \Rightarrow (7*2) \% 10 \neq 1$

$b = 3 \Rightarrow (7*3) \% 10 == 1$

$b = 4$

$b = 5$

$b = 6$

$b = 7$

$b = 8$

$b = 9$

$a = 6$, $m = 7$, $\gcd(a, m) = 1$ $\quad b = a^{-1} \% m = \underline{6}$.

$b = 1 \Rightarrow (6 * 1) \% 7 \neq 1$

$b = 2 \Rightarrow (6 * 2) \% 7 \neq 1$

$b = 3 \Rightarrow (6 * 3) \% 7 \neq 1$

$b = 4 = (6 * 4) \% 7 \neq 1$

$b = 5 \Rightarrow (6 * 5) \% 7 \neq 1$

$b = 6 \Rightarrow (6 * 6) \% 7 = 1$

$a = 6$, $m = 4$ $\quad \gcd(a, m) \neq 1$

$b = 1 \Rightarrow (6 * 1) \% 4 \neq 1$

$b = 2 \Rightarrow (6 * 2) \% 4 \neq 1$

$b = 3 \Rightarrow (6 * 3) \% 4 \neq 1$

$b = 4 \Rightarrow (6 * 4) \% 4 \neq 1$

} Inverse modulo doesn't exist.

Q) Given a, m, $\gcd(a, m) = 1$. Find value of $a^{-1} \% m = ?$ $\quad b = ? \ [1, m-1]$

pseudo-code.

```
for( int i = 1 ; i < m ; i++) {
    if ((a * i) % m == 1) {
        return i
    }
}
```

T.C → O(m)
S.C → O(1)

[// Extended Euclidean → Advance Competitive Programming.]

↳ T.C → O(log m) → very very very tricky.

If m is prime → [Fermat's little theorem.]

$$(a^{m-1}) \,\%\, m = 1$$

// Multiply both sides by $a^{-1} \,\%\, m$

$$(a^{m-1}) \,\%\, m * (a^{-1}) \,\%\, m = a^{-1} \,\%\, m$$

$$(a^{m-1} * a^{-1}) \,\%\, m = a^{-1} \,\%\, m$$

$$(a^{m-2}) \,\%\, m = a^{-1} \,\%\, m$$

↓

- → fast exponentiation
- → power function
- → T.C → log m.

Remember this $\{ a^{-1} \,\%\, m = a^{m-2} \,\%\, m \}$

idea → a = "100" ⇒ 0 0 1
b = "11" 1 1
1 1 1.

→ { Reverse then add char by char }

→ { finally reverse entire ans string }