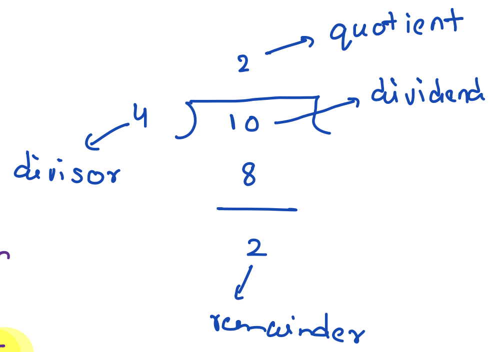
Agenda

1) Modular arithmetic basics (%)

2) what is subsequence and subset, solving questions.

$a \% m \Rightarrow$ remainder when a is divided by m.

$10 \% 4 = 2$



dividend = divisor * quotient + rem

$$\text{rem} = \text{dividend} - \underbrace{\text{divisor} * \text{quotient}}_{\text{gretest multiple of divisor} <= \text{dividend}}$$

$$47 \% 6 = 47 - (\overset{42}{\text{gretest multiple of } 6 <= 47}) = 5$$

$$38 \% 7 = 38 - (\overset{35}{\text{gretest multiple of } 7 <= 38}) = 3$$

$$-47 \% 6 = -47 - (\overset{-48}{\text{greatest multiple of } 6 <= -47})$$

$$= -47 - (-48) = 1$$

$-50 \% 6 = -50 - (-54) = 4$

$-43 \% 7 = -43 - (-49) = 6$

	python		Java | c++
$-47 \% 6$	1	←	$-5 + 6 = 1$
$-50 \% 6$	4	←	$-2 + 6 = 4$
$-43 \% 7$	6	←	$-1 + 7 = 6$

```
if (rem < 0) {
    rem += divisor;
}
```

$ans \,\%\, (10^9 + 7)$

↳ to manage ans. in case of overflow.

$$\left.\begin{matrix}-\infty \\ \\ \\ \infty\end{matrix}\right\} \,\%\, 8 \quad = \quad \boxed{0 \text{ to } 7}$$

$$\left.\begin{matrix}-\infty \\ \\ \\ \infty\end{matrix}\right\} \,\%\, p \quad = \quad \boxed{0 \text{ to } p-1}$$

Properties of modulo

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

$21 \% 4$

$= 1$

$(6 \% 4 + 15 \% 4) \% 4$

$(2 + 3) \% 4$

$= 1$

$a = 6$

$b = 15$

$m = 4$

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

$90 \% 4$

$= 2$

$(6 \% 4 * 15 \% 4) \% 4$

$(2 * 3) \% 4$

$= 2$

$a = 6$

$b = 15$

$m = 4$

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

$-9 \% 4$

$= 3$

$(6 \% 4 - 15 \% 4 + 4) \% 4$

$= (2 - 3 + 4) \% 4$

$= 3 \% 4 = 3$

$a = 6$

$b = 15$

$m = 4$

Q.1 Given a, n, p. Calculate $a^n \% p$.

$1 <= a <= 10^9$

$1 <= n <= 10^5$

$1 <= p <= 10^9$

a = 3 n = 4 p = 5

$(3^4) \% 5 = 1$

$a = 10^9$ n = 3 p = 1003

$(10^9)^3 = 10^{27}$

```
int    solve ( int a, int n, int p) {
    long ans= 1;

     for (int i=1; i<=n; i++) {
               ans=(ans * a) % p;
     }                 ↓      ↓
                      10^9   10^9

     return (int)(ans % p);
}
```

when you are doubtful to apply

mod or not → always apply it.

why ans % p

wrong ans
- overflowed
- logically wrong

ans % p

wrong ans → logically wrong

Subsequence : By removing 0 or more elements.

$$A = \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \\ [3 & 2 & 4 & -1 & 9 & 10] \end{array}$$

✗	✗	✓	✗	✓	✗	{4 9}
✓	✗	✗	✓	✗	✓	{3 -1 10}
✗	✗	✗	✗	✗	✗	{ }
3	✗	✗	✗	✗	✗	{3}

1) continuity does not matter

2) order of indexing matters.

$$A = \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \\ [3 & 2 & 4 & -1 & 9 & 10] \end{array}$$

valid subseq

3 2 4 -1 → ✓

3 4 -1 9 10 → ✓

{ } → ✓

3 9 4 → ✗

A = 3 1 2 $\xrightarrow{\text{Sort}}$ A = 1 2 3

{ }	{ }
{3}	{3}
{1}	{1}
{2}	{2}
{3 1}	{1 3} *
{3 2}	{2 3} *
{1 2}	{1 2}
{3 1 2}	{1 2 3} *

Subset : Same as subsequence but but order of indexing does not matter.

A = 3 1 2 $\xrightarrow{\text{Sort}}$ A = 1 2 3

{ }	{ }
{3}	{3}
{1}	{1}
{2}	{2}
{3 1}	{1 3}
{3 2}	{2 3}
{1 2}	{1 2}
{3 1 2}	{1 2 3}

all subsets of [3 1 2] and [1 2 3] are same.

$$A = [\overset{0}{3}\quad \overset{1}{2}\quad \overset{2}{4}\quad \overset{3}{-1}\quad \overset{4}{9}\quad \overset{5}{10}]$$

3 2 4 10 subseq

3 9 4 subset

count of subsequence

$$A = [\overset{0}{3}\quad \overset{1}{9}\quad \overset{2}{8}\quad \overset{3}{2}]$$

$2 \times 2 \times 2 \times 2 = 16$

total subsequence = 2^n

$\underbrace{\text{total subset}}$ = 2^n

↳ all array elements are distinct.

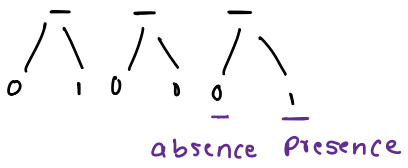
Q-1 Given an array (distinct elements), find if there is any subset with sum = K.

A = [2 7 -1 5 6]   K = 7

{7}   {2 5}   {2 -1 6}   true

A = [3 2 4]

   0 1 2



		2	1	0		
0	→	0	0	0	→	{ }
1	→	0	0	1	→	{3}
2	→	0	1	0	→	{2}
3	→	0	1	1	→	{3 2}
4	→	1	0	0	→	{4}
5	→	1	0	1	→	{3 4}
6	→	1	1	0	→	{2 4}
7	→	1	1	1	→	{3 2 4}

```
boolean subsetSum (int [] A , int k) {
    int n= A.length;
    int tcs = Math.pow(2,n);

    for(int x=0; x < tcs; x++) {
        // check bits of x and
           build your subset sum
        int sum=0;
        for(int i=0; i<n; i++) {
            if (checkbit (x,i) == true) {
                sum+= A[i];
            }
        }
        if (sum == k) {
            return true;
        }
    }
    return false;
}
```

k = 6

A = [4 1 -1 3]

n = 4 , tcs = 16

x	i	sum
0 (0000)	0 to 3	0
1 (0001)	0 to 3	4
2 (0010)	0 to 3	1
⋮		
9 (1001)	0 to 3	7
⋮		
13 (1101)	0 to 3	6

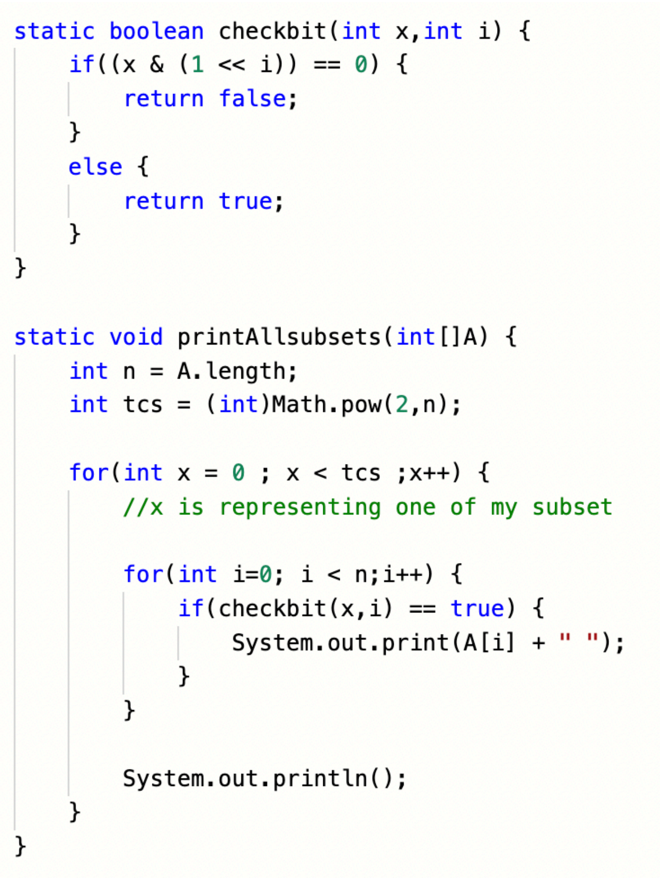
↓ return true

TC: $O(2^n \times n)$

SC: $O(1)$

TODO: Print all subsets

print All subsets



```
static boolean checkbit(int x,int i) {
    if((x & (1 << i)) == 0) {
        return false;
    }
    else {
        return true;
    }
}

static void printAllsubsets(int[]A) {
    int n = A.length;
    int tcs = (int)Math.pow(2,n);

    for(int x = 0 ; x < tcs ;x++) {
        //x is representing one of my subset

        for(int i=0; i < n;i++) {
            if(checkbit(x,i) == true) {
                System.out.print(A[i] + " ");
            }
        }

        System.out.println();
    }
}
```

9 to 10:30

10:30 discussion

Doubts

A = 4 2 3 3
0 1 2 3

K = 6

x = 0

x = 1

⋮

x = 3 → 0 0 1 1 → {4 2 3

⋮

x = 12 → 1 1 0 0 → {3 3 3

checkbit

n = 1

i = 2

n

1 << 2

3	2	1	0
1	1	0	1
0	1	0	0
0	1	0	0

n = 13

i = 1

n

1 << 1

1		0	1
0	0	1	0
0	0	0	0

A = 3   B = 5

$0 \ \overset{1}{\cancel{0}} \ 0 \ \overset{1}{\cancel{0}} \ 0 \ 0 \ 0$

0 1 0 1 0 0 0

$\Rightarrow 2^3 + 2^5 = 8 + 32 = 40$

ans: $2^A + 2^B$

reverse bits

$\begin{array}{llllllll} 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{array}$ 8 bit :

0 → on , ans set 7th bit on

1 → on , ans set 6th bit on

ans: $\overset{1}{\cancel{0}} \ \overset{1}{\cancel{0}} \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$

{ if ith bit is on in a then set
32 - i - 1 bit on in ans.

8bit : 0 0 1 0 1 0 0 1

ans: 1 0 0 1 0 1 0 0

8 - i - 1

8 - 5 - 1 = 2