Dynamic Programming.

- sub-problem (overlapping subproblem)
- Optimal substructure
- Base case

* DP is a brute force with memoization.
* "Trade space for time"

Problem 1: The Triangle (IOI 1994)

```
7
3 8
8 1 0
2 7 4 4
4 5 2 6 5
```

1

→ 7 7

→ 3 10   8 15

→ 8 18   1 16   0 15

2 20   7 25   4 20   4 19

4 24   5 30   2 27   6 26   5 24   ← max

	1	2	3	4	5
vec1	~~0~~ 10	~~0~~ 15	0	0	0
vec2	7				

source-target

```
for ( i = 0; i < N; i++ )
{
    if ( i is even) vec1 is source
                    vec2 "   "
    else
    :
}
```

sol

0						
1						

```
for ( i = 0; i < N; i++ )
{
    source = i % 2;
    target = (i+1) % 2;
```

Problem 2: Ugly Numbers

Ugly numbers are numbers whose only prime factors are 2, 3 or 5. The sequence 1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, … shows the first 11 ugly numbers. By convention, 1 is included. Given a number N (1 <= N <= 10,000). Find the Nth Ugly number. It is guaranteed that output will fit in a 64-bit integer.

Sol 1: Use an array large enough for the Nth ugly number.

Sample Input

7

2, 3, 5

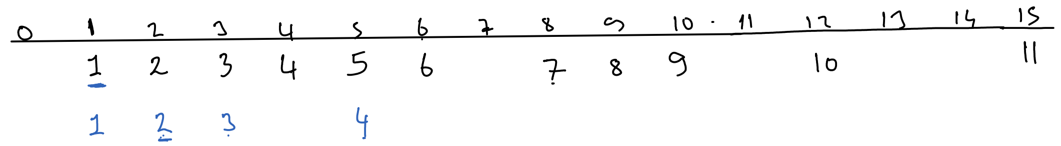
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	2	3	4	5	6		7	8	9		10			11

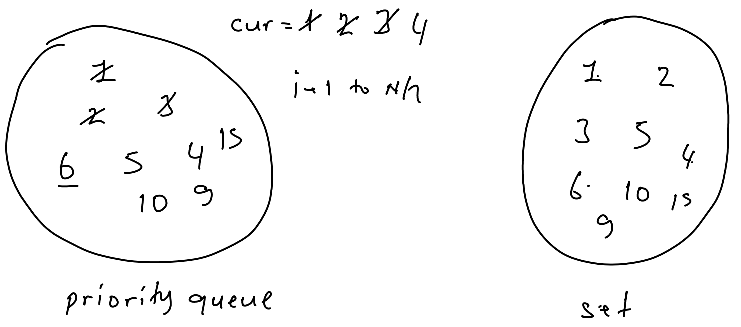
bit integer.

Sample Input
7

Sample Output
8

2,3,5





$O(N \log N)$ time

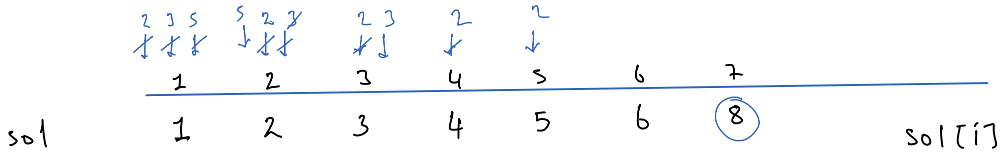
space ?

N = 7

priority queue

set

Sol 2: Generate the first N ugly numbers directly.



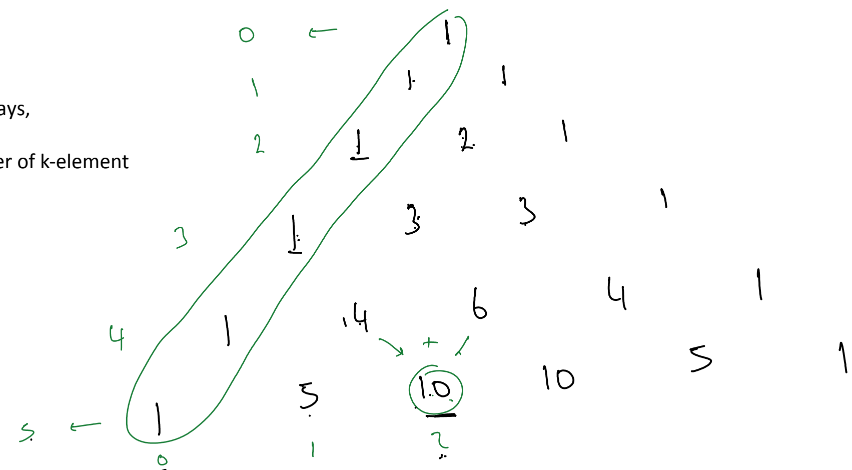
sol[i] is the i-th ugly number.

2, 3, 5

$O(N)$ time and $O(N)$ space

Problem 3: Combination (Pascal) Triangle

A binomial coefficient C(n, k) also gives the number of ways, disregarding order, that k objects can be chosen from among n objects; more formally, the number of k-element subsets (or k-combinations) of an n-element set.
For example, C(4, 2) is 6 and C(5, 2) is 10.



$$C^n_r = \frac{n!}{r!\,(n-r)!}, \quad C^5_2 = \frac{5!}{2!\cdot 3!} = \frac{5\cdot 4\cdot \cancel{3!}}{2\cdot \cancel{3!}} = 10$$

$$C(n,r) = \begin{cases} 1 & \text{if } r == 0 \\ C(n-1,r-1) + C(n-1,r) \end{cases}$$

$$C(n,r) = C(n, n-r)$$

$O(nr)$ time and $O(r)$ space

Combinations with Fermat's Little Theorem

$$C(n,r) \% P \Rightarrow \frac{n!}{r!\,(n-r)!} \% P \qquad \frac{(n! \% P)}{(r! \% P) \cdot ((n-1)! \% P)} \;?\qquad \frac{}{a} \quad a^{-1}$$

$a^{p} = a \pmod p$ if p is prime. $a^{p} - a$ is an integer multiple of p.

$$a^{p} = a^{1} \pmod p \Rightarrow a^{p-1} = 1 \pmod p \Rightarrow a^{p-2} = a^{-1} \pmod p$$

$$\frac{(n! \% P)}{(r! \% P)((n-r)! \% P)} = (n! \% P) \cdot (r!^{p-2} \% P) \cdot (n-r)!^{p-2} \% P$$

$$x^{y} \% P \Rightarrow x^{y} = x \cdot x^{y-1}$$

$$\rightarrow x^{y} = (x^{2})^{y/2} \quad \text{if } y \text{ is even}$$

Divide and conquer

$O(\lg y)$

```
long long powerMod(long long x, long long y, long long p)
{
    if (y % p == 0)
        return 0;
    long long res = 1;

    while (y > 0)
    {
        if (y % 2 == 1)
        {
            res = res * x;   res = res % p;
            ~~y = y-1;~~
        }
        // y is even

        x = x * x;
        x = x % p;
        y = y / 2;
    }
    return res;
```

3

Problem 4: Teamwork (USACO 2018 Dec, Gold)

N=7, K=3

1 ≤ N ≤ 10^4
1 ≤ K ≤ 10^3
SAMPLE INPUT:
7 3
1
15
7
9
2
5
10

SAMPLE OUTPUT:
84

Array partitioning problem. Max partition len is 3.

- Maximize sum of partition scores. One partition score is (max val) x len

3x15 + 9 + 3x10 = 84

	0	1	2	3	4	5	6	7
number	0	1	15	7	9	2	5	10
sol	0	1	~~16~~ 30	~~37~~ ~~31~~ 45	54 ~~48~~ ~~46~~	~~56~~ 63 ~~57~~	~~68~~ ~~64~~ 72	~~82~~ ~~83~~ 84

O(N) space, O(NK) time if you get max in O(1)

Cow Tower (Starleague Training Problem)

1 <= N <= 10,000

SAMPLE INPUT (file tower)

10
2 2
4 17
7 18
7 15
3 14
1 2
6 15
4 12
3 14
3 5

SAMPLE OUTPUT

5

cow1: $w_1 = 10$, $p_1 = 20$

cow2: $w_2 = 5$, $p_2 = 15$

How to sort?
- ~~weight~~
- power
- residual ~~power~~

+5
10 (15, 5) cow2
5 (20, 15) cow1

0
cow1 (20, 10) 10
cow2 (15, 15) 0 ← strongest cow

(power, weight)

1	2	3	4	5	6	7	8	9	10
(2,1)	(2,2)	(5,3)	(12,4)	(14,3)	(14,3)	(15,6)	(15,7)	(17,4)	(18,7)

(3,7) (4,11)

(4,10) (5,14)

(hight, total weigh)

(1,1) (2,4)
(3,~~8~~) (3,7) (4,~~11~~)
(4,10) (5,14)

solution bag (set)

tall	0	1	2	3	4	5	6	7	8	9
	(2, 1)	(2, 7)	(5, 3)	(12, 4)	(14, 3)	(14, 3)	(15, 6)	(15, 7)	(17, 4)	(18, 7)
weight	0	7	4	~~8~~ 7	~~11~~ 10	14				