Getting started with Competitive Programming

Week 10 – Dynamic Programming

Problem 1: Jumping stones

There are n stones in a row from left to right. You are standing on the first stone. From every step from stone number i, you can jump at most k stones further (i+1, i+2, …, i+k). You cannot jump over stone number n. How many ways are there to travel to stone number n?

Input format

You are given two space separated integers on one line, denoting n and k, respectively.

Output format

Print the answer modulo 10^9 + 7.

See the sample tests for sample inputs and outputs.

Problem 2: Knapsack for all subsets

Given are a sequence of N positive integers $A_1,A_2,\ldots,A_N$ and another positive integer S.
For a non-empty subset T of the set$\{1,2,\ldots,N\}$, let us define $f(T)$ as follows:

$f(T)$ is the number of different non-empty subsets $\{x1,x2,\ldots,xk\}$of T such that

- $A_{x1}+A_{x2}+\ldots+A_{xk}=S$.

Find the sum of $f(T)$ over all $2N-1$ subsets$T$ of$\{1,2,\ldots,N\}$. Since the sum can be enormous, print it modulo 998244353.

Constraints

- All values in input are integers.
- $1\leq N\leq 3000$
- $1\leq S\leq 3000$
- $1\leq A_i\leq 3000$

Input

Input is given from Standard Input in the following format:

```
N S

A1,A2,... AN
```

Output

Print the sum of $f(T)$ modulo 998244353.

Sample Input 1

```
3 4

2 2 4
```

Sample Output 1

```
6
```

For each$T$, the value of $f(T)$ is shown below. The sum of these values is 6.

- $f(\{1\})=0$
- $f(\{2\})=0$
- $f(\{3\})=1$ (One subset$\{3\}$satisfies the condition.)
- $f(\{1,2\})=1$ ($\{1,2\}$)
- $f(\{2,3\})=1$ ($\{3\}$)
- $f(\{1,3\})=1$ ($\{3\}$)
- $f(\{1,2,3\})=2$ ($\{1,2\},\{3\}$)

Sample Input 2

```
5 8

9 9 9 9 9
```

Sample Output 2

```
0
```

Sample Input 3

```
10 10

3 1 4 1 5 9 2 6 5 3
```

Sample Output 3

```
3296
```