Perfect Superstring

Perfect Superstring

Satria is a smart person, after he understand the system of binary digits he immediately write a new challenge! The challenge is, give him a binary string S he will tell you a binary string B which is not substring of S. Because Satria is smart, he will always tell buinary string B with minimum length. But everything has a limit, even Satria's intelligence, his brain will be tired if he check subtring of S of length more than N. So he can't tell the binary string B if the binary string S containing all the possible B of length ≤N, in orther word S is superstring of all possible binary string B of length ≤N, lets call S "Perfect Superstring" of order N if it meet this consition.

Gunawan know this challenge and Satria's intelligence limit want to make a binary string S which is perfect superstring of order N. Then he write down all possible binary string B, and concatenate them. For example if he want to make perfect superstring of order 2, he will write: {0,1,00,01,10,11} and concatenate to 0100011011 and submit it. Unfortunately Satria is so smart so his limit will be high. So the Gunawan method to generating perfect superstring of order high is inefficient, but Gunawan never give up, he insist to write his perfect superstring using his method.

After Gunawan writing K digits of his binary string, Tjandra who is curious about Gunawan's serous face ask him why he is so serious.

Tjandra said: "What are you doing?"
Gunawan replied: "I want to solve Satria's challange"
Tjandra replied: "Sound interesting, can you explain the challenge?"
(Gunawan explaining Satria's challenge)
Tjandra said: "wow, that's interesting, how do you solve this challenge?"
(Gunawan explaining his method)
Tjandra said: "That's inefficient, it's enough to concatenate possible binary string B of length N, forget about binary string that has length less than N because all of them will eventually be substring of binary string which has large length, for example if you want to make perfect superstring of order 2 your string will be 0100011011, but it's enough to just concatenate {00,01,10,11} and become 00011011, it's shorter and it's perfect superstring of length 2!"
Gunawan replied: "wow it's shorter! I never think about that, is that the minimum possible length"
Tjandra replied: "No, there are the shorter one, for example in binary string 000110011 the substring 11 appear 2 times, so we can delete one digit of '1's and i't still meet the perfect superstring of order 2"

Gunawan replied: "You're right, I don't like wasting my energy, I want the perfect superstring of order N with minimum possible length!"

Tjandra replied: "Unfortunately no simple method for generating the minimum posible length of perfect superstring, but I know someone who can generate it for you with modern computer" (Tjandra pulled his old phone from his pocket, getting ready to call you for help)

Gunawan said: "Wait, I already write K binary digits, I don't want to delete them and start from the beginning"

Tjandra replied: "What -_- Ok, I'll consider that"

Now Tjandra call you using his old phone for help, because he know you're skilled at string processing. Can you help them?

Input

The first line there is an integer N denoting the order of perfect superstring on Satria's challenge. On the second line there are one integer K and one string X, K denoting the length of binary string that is already writen by Gunawan, and X is that binary string itself.

Output

The the first line you should output minimum possible length of {perfect superstring of order N starting with X of length K}, let's call this number L.

On the second line you should output a binary string of length L starting with K and it is perfect superstring of order N.

Constraint

$1 \le N \le 20$

$0 \le K \le N$

String X will have length K, in order word |X|=K. It can be empty string if K=0.

Sample 1

Input

```
1
1 0
```

Output

```
2
01
```

Sample 2

Input

```
1
1 1
```

Output

2
10

Sample 3

2
2 01

5
01100

Sample 4

3
3 110

10
1101000111

Sample 5

3
3 110

10
1100010111

Sample 4 (and 5) Explanation

In sample 4, string 1101000111 starting with string 110: [110]1000111, so Gunawan doesn't need to erase his already written number from the begining.

It also meet the perfect supertring of order 3 because all binary string of length ≤3 is substring of that string.

Here is the proof:

0: 11**0**1000111
1: **1**101000111
00: 1101**00**0111
01: 11**01**000111
10: 1**10**1000111
11: **11**01000111

000: 1101**000**111
001: 11010**001**11
010: 11**010**00111
011: 110100**011**1
100: 110**100**0111
101: 1**101**000111
110: **110**1000111
111: 1101000**111**

Note that there can be multiple solution, the string 1100010111 as shown on sample 5 is another minimal length perfect superstring of of order 3 starting with 110.