1992 ACM MID-CENTRAL REGIONAL

PROGRAMMING CONTEST

Problem #3 - Digit Twirling

Program File: `TWIRL.PAS` or `TWIRL.C`
Input File: `TWIRL.IN`
Output File: `TWIRL.OUT`

This problem deals with the manipulation of base-n integers. Given a number in a particular base, you are to rotate the digits some number of positions, and then convert it to a target base. We will call this operation "twirling". This might sound useless. Well, it is. However, it could prove to be challenging.

The input file contains several digit twirling problems, one problem on each line. Each problem consists of four parameters: B_1, N, R, and B_2. These parameters are to be interpreted as follows:

- B_1 is an integer that represents the starting base ($2 \le B_1 \le 36$). Capital letters will be used as digits when B_1 exceeds 10. For example 'F' is the digit 15 when the base is 16 or higher, and 'Z' is the digit 35 when the base is 36.
- N is the base-B_1 integer that is to be "twirled". N is non-negative and will contain from 1 to 30 digits.
- R is an integer (always base-10) that represents the number of digit positions that the number N should be rotated. If R is positive the rotation should be to the left. If R is negative the rotation should be to the right. For example, a rotation of 2 for the number 12345 yields 34512; while a rotation of -1 for the number 12345 yields 51234. R may be any valid standard integer.
- B_2 is an integer that represents the ending base ($2 \le B_2 \le 36$). Your answer should be displayed using this base. If the $B_2 > 10$, use capital letters to represent the digits 10 through (B_2-1).

These four parameters will be separated from each other by 1 space. The end of the file will be marked with a line that contains 4 zeros: `0 0 0 0`.

The output file, `TWIRL.OUT`, should contain one line for each problem represented in the input file. The output should first echo (within brackets) the four parameters read from the input file followed by an equal sign and the answer. You may assume that no answer will require more than 30 digits. Use the format:

`{B1 N R B2} = answer`

For example if the input file contains the lines:

```
10 12345 -2 2
36 ZA 1 10
0 0 0 0
```

the output file should contain the lines:

```
{10 12345 -2 2} = 1011000001000011
{36 ZA 1 16} = 18B
```

The answer for the first problem is found by rotating 12345 two digits to the right which yields 45123. The value 45123 in base-2 is 1011000001000011.

The answer for the second problem is found by rotating ZA 1 digit to the left which yields AZ. The value AZ in base-16 is 18B.