Digital Calculator

My brother John was so fond of his digital calculator that when the one I had once given him as present broke down, he was very depressed. So, I promised to develop him a program that would actually efficiently simulate such a digital calculator on the computer screen. Can you help me develop such a program?

Task

Your task is to develop a program that can efficiently simulate a digital calculator. The program should be able to receive as input from the standard input stream:

1. Two integer numbers
2. An arithmetic operator (i.e. + for addition, - for subtraction, \* for multiplication, / for the **integral (Euclidean) division**, or % the for the remainder of the integral(Euclidean) division)
3. A character used to draw the numbers on the screen
4. A positive odd integer value representing the size of each digit
5. A positive integer value representing the gap size between the digits

and then draw the arithmetic calculation and the result on the standard output stream. An example output of the program is provided at Fig.1, for the arithmetic operation (7 \* -22), using the character '@' to represent digits, selecting a size value equal to 5 and a gap size equal to 3 (dots stand for spaces).

................................@@@@@

...................................@.

..................................@..

.................................@...

................................@....

.....................................

........................@@@@@...@@@@@

.\*\*\*........................@.......@

.\*\*\*............@@@@@...@@@@@...@@@@@

.\*\*\*....................@.......@....

........................@@@@@...@@@@@

.....................................

-------------------------------------

.....................................

..................@.....@@@@@...@...@

.................@@.....@.......@...@

........@@@@@.....@.....@@@@@...@@@@@

..................@.........@.......@

................@@@@@...@@@@@.......@

Fig. 1. An example of the program output for the arithmetic operation (7 \* -22), using the character '@' to represent digits, selecting a size value equal to 5 and a gap size equal to 3

Input

The program should receive as input:

1. One integer value (-46340 ≤ ***num1*** ≤ 46340) representing the first operand
2. Another integer value (-46340 ≤ ***num2*** ≤  46340) representing the second operand
3. One of the characters +, -, \*, /, % representing the operator that will be applied to the operands.
4. One character used for the drawing of the operands.
5. One positive **odd** integer value (5 ≤ ***size*** ≤ 31) representing the size (i.e. width and height) of each digit
6. Another positive integer value (1 ≤ ***gaps*** ≤ 100) representing the gap size between the digits.

These parameters will be received by the standard input stream, one parameter per line and according to the order described above.

Output

The program should output on the standard output stream the arithmetic calculation between the input operands as well as the calculated result. More specifically:

1. At the first ***size*** lines, the first integer (num1) number should be drawn
2. Then an empty line (i.e. ***full of spaces***) should be drawn
3. At the following ***size*** lines, the second integer (num2) number should be drawn.
4. Again, an empty line (i.e. ***full of spaces***) should be drawn
5. Then, a line ***full of dashes (-)*** should be drawn, followed by an empty line (i.e. ***full of spaces***)
6. Finally, at the last ***size*** lines, the result of the arithmetic operation should be drawn.

With regards to the horizontal alignment of the output, right side justification should be used. In particular:

1. The first ***size*** columns correspond to the operator of the calculation
2. Then ***gaps*** columns will be left blank (i.e. full of spaces)
3. Then, the digits of each operand as well as the ones of the deriving result should be drawn using (***size*** + ***gaps***) columns per digit and aligning the digits to the right side (see Fig. 1)

Each digit, should be drawn using the character that was received as input, whereas the operator should be drawn using the symbol that corresponds to the selected mathematical operation (e.g. the '+' should be used to draw the addition operand, whereas the '%' symbol should be used if the remainder operation was selected). The following figure (Fig. 2) graphically depicts all the digits (using the symbol '#') and all possible operators (using the symbol that corresponds to each mathematical operation).

.........#....#####..#####..#...#..#####..#####..#####..#####..#####..#####

........##........#......#..#...#..#......#.........#...#...#..#...#..#...#

#####....#....#####..#####..#####..#####..#####....#....#####..#####..#...#

.........#....#..........#......#......#..#...#...#.....#...#......#..#...#

.......#####..#####..#####......#..#####..#####..#......#####..#####..#####

.................................

........................./..%...%

..+............\*\*\*....../......%.

+++++..-----...\*\*\*...../......%..

..+............\*\*\*..../......%...

...................../......%...%

(a)

............#.....#######..#######..#.....#..#######..#######..#######..#######..#######..#######

...........##...........#........#..#.....#..#........#.............#...#.....#..#.....#..#.....#

..........#.#...........#........#..#.....#..#........#............#....#.....#..#.....#..#.....#

#######.....#.....#######..#######..#######..#######..#######.....#.....#######..#######..#.....#

............#.....#..............#........#........#..#.....#....#......#.....#........#..#.....#

............#.....#..............#........#........#..#.....#...#.......#.....#........#..#.....#

.........#######..#######..#######........#..#######..#######..#........#######..#######..#######

...........................................

................................./..%.....%

...+...............\*\*\*\*\*......../........%.

...+...............\*\*\*\*\*......./........%..

+++++++..-------...\*\*\*\*\*....../........%...

...+...............\*\*\*\*\*...../........%....

...+...............\*\*\*\*\*..../........%.....

.........................../........%.....%

(b)

...............#......#########..#########..#.......#..#########..#########..#########..#########..#########..#########

..............##..............#..........#..#.......#..#..........#.................#...#.......#..#.......#..#.......#

.............#.#..............#..........#..#.......#..#..........#................#....#.......#..#.......#..#.......#

............#..#..............#..........#..#.......#..#..........#...............#.....#.......#..#.......#..#.......#

#########......#......#########..#########..#########..#########..#########......#......#########..#########..#.......#

...............#......#..................#..........#..........#..#.......#.....#.......#.......#..........#..#.......#

...............#......#..................#..........#..........#..#.......#....#........#.......#..........#..#.......#

...............#......#..................#..........#..........#..#.......#...#.........#.......#..........#..#.......#

...........#########..#########..#########..........#..#########..#########..#..........#########..#########..#########

.....................................................

........................................./..%%......%

....+..................\*\*\*\*\*\*\*........../...%%.....%.

....+..................\*\*\*\*\*\*\*........./..........%..

....+..................\*\*\*\*\*\*\*......../..........%...

+++++++++..---------...\*\*\*\*\*\*\*......./..........%....

....+..................\*\*\*\*\*\*\*....../..........%.....

....+..................\*\*\*\*\*\*\*...../..........%......

....+..................\*\*\*\*\*\*\*..../..........%.....%%

................................./..........%......%%

(c)

Fig. 2. Representation of the digits 0-9 (using the symbol '#') and all possible operators (using the symbol that corresponds to each mathematical operation) for size=5 (case a), size=7 ( case b) and size=9( case c).

Although the representation of the majority of the aforementioned digits and operators seems to be quite straightforward, special attention has to be paid when drawing:

1. The digit 1, where the diagonal slope should have a height equal to (***size***/2).
2. The digit 7, which is formed out of an horizontal line and a diagonal slope
3. The operator +, which have a height equal to (***size***-2)
4. The operator  \*, which have a height ***and width*** equal to (***size***-2)
5. The operator %, where each small circle is represented by a square with edge length equal to (***size***/4)

*Note: There is a newline character at the end of the last line of the output.*

Sample Input 1

-3

-6

\*

@

7

2

Sample Output 1

@@@@@@@

@

@

@@@@@@@ @@@@@@@

@

@

@@@@@@@

@@@@@@@

\*\*\*\*\* @

\*\*\*\*\* @

\*\*\*\*\* @@@@@@@ @@@@@@@

\*\*\*\*\* @ @

\*\*\*\*\* @ @

@@@@@@@

-------------------------

@ @@@@@@@

@@ @ @

@ @ @ @

@ @@@@@@@

@ @ @

@ @ @

@@@@@@@ @@@@@@@

Sample Input 2

30120

1215

%

$

5

3

Sample Output 2

$$$$$ $$$$$ $ $$$$$ $$$$$

$ $ $ $$ $ $ $

$$$$$ $ $ $ $$$$$ $ $

$ $ $ $ $ $ $

$$$$$ $$$$$ $$$$$ $$$$$ $$$$$

% % $ $$$$$ $ $$$$$

% $$ $ $$ $

% $ $$$$$ $ $$$$$

% $ $ $ $

% % $$$$$ $$$$$ $$$$$ $$$$$

---------------------------------------------

$$$$$ $$$$$ $$$$$

$ $ $ $ $

$$$$$ $$$$$ $ $

$ $ $ $ $

$$$$$ $$$$$ $$$$$