Spreadsheet Simulator

Need to develop a simple batch spreadsheet processor. It must be able to process cells just like in a ordinary spreadsheet, but will use simplified expressions. Each cell may contain:

- Nothing
- Non-negative integer number.
- Text labels, which is started with ' symbol.
- expression, which is started with '=' symbol and may contain non-negative integers, cell references, and simple arithmetic operations. There are no parenthesis allowed, and all operations have equal priority. References consist of single letter followed by single digit.

These restrictions has been made in order to simplify parsing, because parsing implementation is not important part of these problem. You should safely rely on these restrictions. Here is the grammar for cells content:

```
<expression> = ‘=’ <term> { <operation> <term> }
<term> = <cell reference> | <non-negative number>
<cell reference > = <letter> <digit>
<operation> = ‘+’ | ‘-‘ | ‘*’ | ‘/’
<text> = ‘’’{<printable character>}
```

Sheet processing rules are:

- all expressions should be substituted with calculated results.
- all calculations should be taken with signed integer arithmetic.
- text cells should be evaluated to corresponding text without ' prefix.
- There are no operations on text labels allowed.
- in case of any error in cell evaluation resulting cell must contain word with error description started with ‘#’ symbol.

Input and Output

Program should take sheet description with formulas from the standard input, evaluate it, and print resulting table to the standard output. Input data represented as tabulation-separated table. First line contains pair of integers separated with tabulation and designating table's height and length. Then, from the next line the cells themselves come, each in the grammar described above.

Sample Input

```
3           4
12          =C2         3           'Sample
=A1+B1*C1/5 =A2*B1      =B3-C3      'Spread
'Test       =4-3        5           'Sheet
```

Sample Output

```
 12          -4         3            Sample
 4          -16        -4            Spread
 Test         1         5             Sheet
```

Solution Guidelines

Production code quality is a must. The shorter and easy to read solution is acceptable. However the solution should leverage OOD patterns as needed and be easily extendible. Any OO programming language will work to develop the task.

Suppose that this problem description define phase #1 of the project. You must implement nothing else than required for it. However, it’s known that there is phase #2 planned, and requirements will be:

- extend formula language to handle text fields.
- optimize for performance on huge tables using multithreaded processing.

You will be asked to describe changes you would made to your solution during the phase #2 during the interview.