In most role-playing games, die rolls required by the system are given in the form `AdX`. `A` and `X` are positive integers, separated by the letter `'d'`, which stands for *die* or *dice*.

- `A` is the number of times the die should be rolled (usually omitted if `1`).
- `X` is the number of faces on the die.

To this basic notation, an additive modifier can be appended that yields expressions in the form `AdX+B` or `AdX-B`. `B` is a number added to the sum of the rolls (or subtracted from it). So, `1d20-10` would indicate a single roll of a 20-sided die with 10 being subtracted from the result.

You're reading the `rules` of a famous *Bugs and Bugfixes* role-playing game involving dice. What is the maximum number of points you can get, assuming that you roll the dice according to each formula present within `rules`?

It is guaranteed that all the formulas that appear in `rules` are correct (i.e. `A` and `X` are always positive in a formula-like substring), and any two substrings that may be formulas do not overlap.

**Example**

For `rules = "Roll d6-3 and 4d4+3 to pick a weapon, and finish the boss with 3d7!"`,
the output should be
`bugsAndBugfixes(rules) = 43`.

There are three formulas in the `rules`.

- `d6-3` indicates a single roll of a 6-sided die, with `3` subtracted from the result. The maximum number that is possible to get is thus `6 - 3 = 3`.
- `4d4+3` stands for `4` rolls of a 4-sided die, with `3` added to the result. It is possible to get `4 * 4 + 3 = 19` points.
- `3d7` means `3` rolls of a 7-sided die. The maximum number to obtain with it is `3 * 7 = 21`.

**Input/Output**

- **[execution time limit] 4 seconds (js)**
- **[input] string rules**

  Rules given as a string.

  *Guaranteed constraints:*
  `1 ≤ rules.length ≤ 100`.

- **[output] integer**

  The maximum possible number of points. If there are no formulas in `rules`, the output should be `0`.

## [JavaScript (ES6)] Syntax Tips

```
// Prints help message to the console
// Returns a string
function helloWorld(name) {
    console.log("This prints to the console when you Run Tests");
    return "Hello, " + name;
}
```