--- Day 19: Monster Messages ---

You land in an airport surrounded by dense forest. As you walk to your high-speed train, the Elves at the Mythical Information Bureau contact you again. They think their satellite has collected an image of a **sea monster**! Unfortunately, the connection to the satellite is having problems, and many of the messages sent back from the satellite have been corrupted.

They sent you a list of **the rules valid messages should obey** and a list of **received messages** they've collected so far (your puzzle input).

The **rules for valid messages** (the top part of your puzzle input) are numbered and build upon each other. For example:

```
0: 1 2
1: "a"
2: 1 3 | 3 1
3: "b"
```

Some rules, like `3: "b"`, simply match a single character (in this case, `b`).

The remaining rules list the sub-rules that must be followed; for example, the rule `0: 1 2` means that to match rule `0`, the text being checked must match rule `1`, and the text after the part that matched rule `1` must then match rule `2`.

Some of the rules have multiple lists of sub-rules separated by a pipe (`|`). This means that **at least one** list of sub-rules must match. (The ones that match might be different each time the rule is encountered.) For example, the rule `2: 1 3 | 3 1` means that to match rule `2`, the text being checked must match rule `1` followed by rule `3` **or** it must match rule `3` followed by rule `1`.

Fortunately, there are no loops in the rules, so the list of possible matches will be finite. Since rule `1` matches `a` and rule `3` matches `b`, rule `2` matches either `ab` or `ba`. Therefore, rule `0` matches `aab` or `aba`.

Here's a more interesting example:

```
0: 4 1 5
1: 2 3 | 3 2
2: 4 4 | 5 5
3: 4 5 | 5 4
4: "a"
5: "b"
```

Here, because rule `4` matches `a` and rule `5` matches `b`, rule `2` matches two letters that are the same (`aa` or `bb`), and rule `3` matches two letters that are different (`ab` or `ba`).

Since rule `1` matches rules `2` and `3` once each in either order, it must match two pairs of letters, one pair with matching letters and one pair with different letters. This leaves eight possibilities: `aaab`, `aaba`, `bbab`, `bbba`, `abaa`, `abbb`, `baaa`, or `babb`.

Rule `0`, therefore, matches `a` (rule `4`), then any of the eight options from rule `1`, then `b` (rule `5`): `aaaabb`, `aaabab`, `abbabb`, `abbbab`, `aabaab`, `aabbbb`, `abaaab`, or `ababbb`.

The **received messages** (the bottom part of your puzzle input) need to be checked against the rules so you can determine which are valid and which are corrupted. Including the rules and the messages together, this might look like:

```
0: 4 1 5
1: 2 3 | 3 2
2: 4 4 | 5 5
3: 4 5 | 5 4
4: "a"
5: "b"

ababbb
bababa
abbbab
aaabbb
aaaabbb
```

Your goal is to determine the number of messages that completely match rule `0`. In the above example, `ababbb` and `abbbab` match, but `bababa`, `aaabbb`, and `aaaabbb` do not, producing the answer `2`. The whole message must match all of rule `0`; there can't be extra unmatched characters in the message. (For example, `aaaabbb` might appear to match rule `0` above, but it has an extra unmatched `b` on the end.)

How many messages completely match rule `0`?

Your puzzle answer was `279`.

--- Part Two ---

As you look over the list of messages, you realize your matching rules aren't quite right. To fix them, completely replace rules `8: 42` and `11: 42 31` with the following:

```
8: 42 | 42 8
11: 42 31 | 42 11 31
```

This small change has a big impact: now, the rules do contain loops, and the list of messages they could hypothetically match is infinite. You'll need to determine how these changes affect which messages are valid.

Fortunately, many of the rules are unaffected by this change; it might help to start by looking at which rules always match the same set of values and how those rules (especially rules `42` and `31`) are used by the new versions of rules `8` and `11`.

(Remember, you only need to handle the rules you have; building a solution that could handle any hypothetical combination of rules would be significantly more difficult.)

For example:

```
42: 9 14 | 10 1
9: 14 27 | 1 26
10: 23 14 | 28 1
1: "a"
11: 42 31
5: 1 14 | 15 1
19: 14 1 | 14 14
12: 24 14 | 19 1
16: 15 1 | 14 14
31: 14 17 | 1 13
6: 14 14 | 1 14
2: 1 24 | 14 4
0: 8 11
13: 14 3 | 1 12
15: 1 | 14
17: 14 2 | 1 7
23: 25 1 | 22 14
28: 16 1
4: 1 1
20: 14 14 | 1 15
3: 5 14 | 16 1
27: 1 6 | 14 18
14: "b"
21: 14 1 | 1 14
25: 1 1 | 1 14
22: 14 14
8: 42
26: 14 22 | 1 20
18: 15 15
7: 14 5 | 1 21
24: 14 1

abbbbbabbbaaaababbaabbbbabababbbabbbbbbabaaaa
bbabbbbaabaabba
babbbbaabbbbbabbbbbbaabaaabaaa
aaabbbbbbaaaabaababaabababbbabaaabbababababaaa
bbbbbbbaaaabbbbaaabbabaaa
bbbababbbbaaaaaaaabbababaaababaabab
ababaaaaaabaaab
ababaaaaabbbaba
baabbaaaabbaaaababbaabbb
abbbbabbbbaaaababbbbbbaaaababb
aaaaabbaabaaaaababaa
aaaabbaaaabbaaa
aaaabbaabbaaaaaaabbbabbbaaabbaabaaa
babaaabbbaaabaababbaababababaaab
aabbbbbaabbbaaaaaabbbbbababaaaaabbaaabba
```

Without updating rules `8` and `11`, these rules only match three messages: `bbabbbbaabaabba`, `ababaaaaaabaaab`, and `ababaaaaabbbaba`.

However, after updating rules `8` and `11`, a total of `12` messages match:

- `bbabbbbaabaabba`
- `babbbbaabbbbbabbbbbbaabaaabaaa`
- `aaabbbbbbaaaabaababaabababbbabaaabbababababaaa`
- `bbbbbbbaaaabbbbaaabbabaaa`
- `bbbababbbbaaaaaaaabbababaaababaabab`
- `ababaaaaaabaaab`
- `ababaaaaabbbaba`
- `baabbaaaabbaaaababbaabbb`
- `abbbbabbbbaaaababbbbbbaaaababb`
- `aaaaabbaabaaaaababaa`
- `aaaabbaabbaaaaaaabbbabbbaaabbaabaaa`
- `aabbbbbaabbbaaaaaabbbbbababaaaaabbaaabba`

After updating rules `8` and `11`, how many messages completely match rule `0`?

```
Your puzzle answer was 384.

Both parts of this puzzle are complete! They provide two gold stars: **

At this point, you should return to your Advent calendar and try another puzzle.

If you still want to see it, you can get your puzzle input.

You can also [Share] this puzzle.
```