--- Day 8: Matchsticks ---

Space on the sleigh is limited this year, and so Santa will be bringing his list as a digital copy. He needs to know how much space it will take up when stored.

It is common in many programming languages to provide a way to escape special characters in strings. For example, C, JavaScript, Perl, Python, and even PHP handle special characters in very similar ways.

However, it is important to realize the difference between the number of characters in the code representation of the string literal and the number of characters in the in-memory string itself.

For example:

- `""` is `2` characters of code (the two double quotes), but the string contains zero characters.
- `"abc"` is `5` characters of code, but `3` characters in the string data.
- `"aaa\"aaa"` is `10` characters of code, but the string itself contains six "a" characters and a single, escaped quote character, for a total of `7` characters in the string data.
- `"\x27"` is `6` characters of code, but the string itself contains just one - an apostrophe (`'`), escaped using hexadecimal notation.

Santa's list is a file that contains many double-quoted string literals, one on each line. The only escape sequences used are `\\` (which represents a single backslash), `\"` (which represents a lone double-quote character), and `\x` plus two hexadecimal characters (which represents a single character with that ASCII code).

Disregarding the whitespace in the file, what is the number of characters of code for string literals minus the number of characters in memory for the values of the strings in total for the entire file?

For example, given the four strings above, the total number of characters of string code (`2 + 5 + 10 + 6 = 23`) minus the total number of characters in memory for string values (`0 + 3 + 7 + 1 = 11`) is `23 - 11 = 12`.

Your puzzle answer was `1342`.

--- Part Two ---

Now, let's go the other way. In addition to finding the number of characters of code, you should now encode each code representation as a new string and find the number of characters of the new encoded representation, including the surrounding double quotes.

For example:

- `""` encodes to `"\"\""`, an increase from `2` characters to `6`.
- `"abc"` encodes to `"\"abc\""`, an increase from `5` characters to `9`.
- `"aaa\"aaa"` encodes to `"\"aaa\\\"aaa\""`, an increase from `10` characters to `16`.
- `"\x27"` encodes to `"\"\\x27\""`, an increase from `6` characters to `11`.

Your task is to find the total number of characters to represent the newly encoded strings minus the number of characters of code in each original string literal. For example, for the strings above, the total encoded length (`6 + 9 + 16 + 11 = 42`) minus the characters in the original code representation (`23`, just like in the first part of this puzzle) is `42 - 23 = 19`.

Your puzzle answer was `2074`.

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.