

The purpose of this project is to use GOLD to define finite state transducers to code and decode input strings. You should hand in the .gold file with your solution containing both the coder and decoder.

Task 1. Build a coder that reads an input made-up of strings of lowercase letters separated by a single space and that end with the symbol \$.

The coder should output breakup strings of length greater than four into substrings of length 4 separated by the symbol @. If the length of the string is not multiple of four, the last substring should be padded with #'s. These substrings are not separated by spaces but with the symbol * preceded by the number of substrings that composed the original string modulo 3. The last substring is followed by this number and then \$.

For example:

1. **input** a bb ccc ddd abcdefg xyzxyzxyzxyzdddd gh\$
output a###1*bb##1*ccc#1*ddd#1*abcd@efg#2*xyzx@yzxy@zxyz@dddd1*gh##1\$
2. **input** try a long word abcdefghijklmnopqrstuvwxyz aha\$
output try#1*a###1*long1*word1*abcd@efgh@ijkl@mnop@qrst@uvw@xyz#1*aha#1\$

Task 2. Build a decoder to decode the strings coded by the coder from the previous exercise. It should verify that the coding process was done correctly: #'s well placed and the numbers should be correct.

Attached you will find a Gold project that implements the zip unzip example seen in class. You would have to modify zip.gold and unzip.gold for the coder and decoder respectively.