Universidad de los Andes
Facultad de Ingeniería

Ingeniería de Sistemas y Computación
202410-ISIS-1106 Lenguajes y Máquinas
Project 2
Finite State Transducers.

Use Gold to implement the coder-decoder described below:

1. Coder: Reads a string of the form: $\omega_1!\,\omega_2!\,\omega_3!\,...!\,\omega_{2n-1}!\,\omega_{2n}\#$ with $n \geq 0$, $\omega_i \in ('a'..'d')^*$ and length 3 (for each $0 \leq i \leq n$) . The automaton should output $\rho_1!\,\rho_2!\,\rho_3!\,...!\,\rho_{2n-1}!\,\rho_{2n}\#$ where for each i between 1 and $2n$: If i is odd then ρ_i is ω_i; if i is even and $\omega_i = \omega_{i-1}$ then ρ_i = "*" otherwise ρ_i is ω_i. The automaton should report an error if the length of any substring is not 3. Note that the letters in each substring do not have to be different.

Input	Output
`abc!abc!aaa!aba!abc!ccc!ccc!dad#`	`abc!*!aaa!aba!abc!ccc!ccc!dad#`
`abc!abc!aaa!aba!abc!ccc!ccc!ccc!dad!ddd#`	`abc!*!aaa!aba!abc!ccc!ccc!*!dad!ddd#`

2. Decoder: Reads the output produced by the previous automaton and outputs the original string. It should verify that substrings in even positions should not be equal to the previous substring. The automaton should also report an error if the length of any substring that is not * is not 3.

We include a Gold project with a coder-decoder. You may modify it to develop your project.