HW 3: More foul logic

Re-submit Assignment

Due Feb 23 by 11:59pm **Points** 100 **Submitting** a file upload **File Types** pl

WHAT TO SUBMIT:

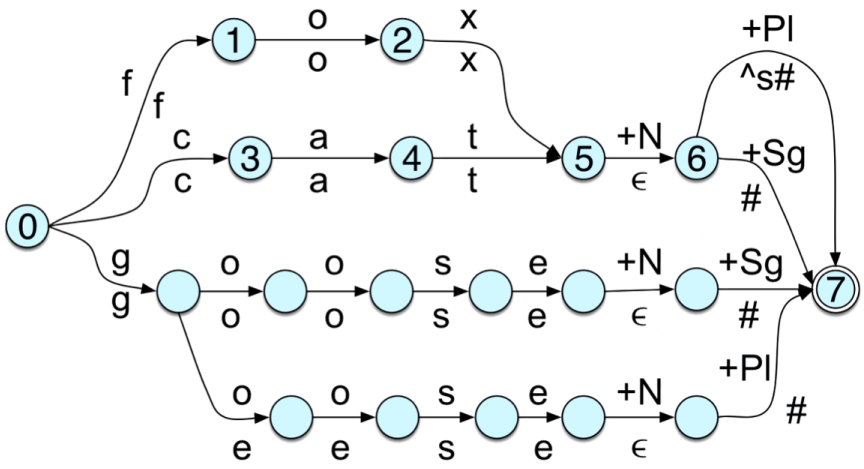
- One file named hw3q1.pl for part 1 (see below).
- One file named hw3q2.pl for part 2 (see below).

(Both files should be prolog files.)

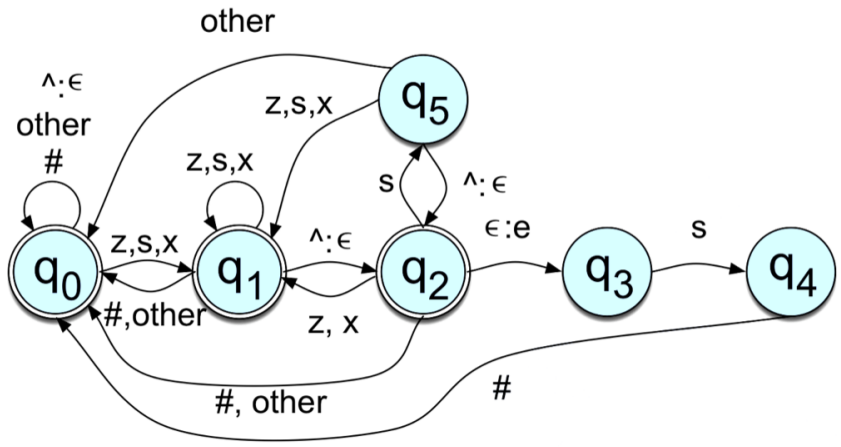
PART 1: More-foul-logical spelling

Consider the FSTs (from Jurafsky & Martin) discussed in class and reproduced below. Together, these two FSTs show how one can perform morphological inflection to express number (singular and plural) in nouns using an approach called two-level morphology.

The first FST is the level where the input has the form WORD+N+Pl, or WORD+N+Sg, where WORD is a noun (fox, cat, goose), +N represents that it is a noun, +Pl denotes plural, and +Sg denotes singular. The output of this FST is an intermediate form used in two-level morphology: the original word with a morpheme (or no visible morpheme) concatenated to it, but using only the canonical forms of the morphemes (i.e. without observing spelling rules or phonological rules).



The second FST takes the intermediate form as input, and produces the *spelling* of the inflected word.



Examine the file hw3q1-template.pl in the HW3 folder, under Files on canvas. It corresponds to the second level FST above. Make sure you understand how the FST above is represented in the file. Notice that there are missing transitions! Your first task is to fill in the missing transitions, and save the complete FST in a file called **hw3q1.pl**. (You don't need to implement the first level FST, just the second level.)

WHAT TO TURN IN FOR PART 1:

Submit a file named hw3q1.pl, which contains the FST from hw3q1-template.pl with your additional transitions to match the figure above.

Your submission should produce appropriate output when given inputs such as:

```
f o x ^ s #
```

```
f o x #
```

```
c a t ^ s #
```

```
g o o s e #
```

And other similar strings that are not included in the first FST above, but follow the same format. This is not an exhaustive list, and your submission will not be evaluated on its ability to produce the correct output alone; your FST must match the picture above.

PART 2: More-foul-logical sounds

Consider now the allomorphs of the plural morpheme in English discussed in class: [z] (as in dogs), [s] (as in cats), [əz] (as in foxes). Let's represent them with the arpabet symbols Z, S and IH Z.

Create an FST that takes as input the *pronunciation* of a noun (represented using arpabet symbols as in the CMU pronunciation dictionary) concatenated with a representation of the plural morpheme and the end of word symbol # used by J&M, and outputs the *pronunciation* of the plural form of the noun, followed by the symbol #.

For example, given the input:

```
d aa g ^ s #
```

(which corresponds to the word dog, the arbitrary symbol ^, the plural morpheme s, and the end of word symbol #) will produce as output

```
d aa g z #
```

(which is the pronunciation of the plural form *dogs*).

Other examples:

Input (cat+s)

```
k ae t ^ s #
```

Output (cats)

```
k ae t s #
```

Input (flash+s)

```
f l ae sh ^ s #
```

Output (flashes)

```
f l ae sh ih z #
```

WHAT TO TURN IN FOR PART 2:

Submit a file named hw3q2.pl, containing your FST as prolog statements.

TIP: In the templates provided, you will notice that the ^ and # symbols appear as '^' and '#' (enclosed by single-quotes). **Make sure to use the single-quotes with these symbols**, but not with arpabet symbols.

AVAILABLE RESOURCES

In the HW3 folder you will find:

- fst.pl: the same FST code we have been using.
- hw3q1-template.pl: a starting point for part 1.
- hw3q2-template.pl: a starting point for part 2.