


```
(0 (0 "dog" "dog"))
```

This transducer only has a single state, which is both start and final. Every time it sees the word "big" on its input, it produces "small" on its output. Every time it sees the word "dog," it outputs "dog." So it will convert an input like "big big dog big" into "small small dog small." Notice that

Much of linguistic theory revolves around dividing strings into two categories: grammatical and ungrammatical. This distinction is useful in real applications, too. For example, if a translator has two possible outputs, we should prefer a grammatical one over an ungrammatical one. A lot of grammatical constraints can be encoded in FSAs and other devices such as context-free grammars (CFGs).


```
%%%%%%%% Filename: wfst1 %%%%%%%%%
S
(S (S "they" "PRO" 1.0))
(S (S "can" "AUX" 0.99))
(S (S "can" "VERB" 0.01))
(S (S "fish" "NOUN" 0.7))
(S (S "fish" "VERB" 0.3))
```

Exercise. Try this command: % carmel -Ok 4 fsa7 wfst1


```
% carmel -Irk 5 wfsa-english wfst-epron wfst-jpron  
wfst-write-katakana fsa-observed
```

[Note: these large automata files are not in the

(No exercises).