Write Out the Word and Reset the FSM

```
*write = 0;
puts (buf);
```
Write current word to **stdout** (with a newline).

```
write = buf;
word_len = 0;
```
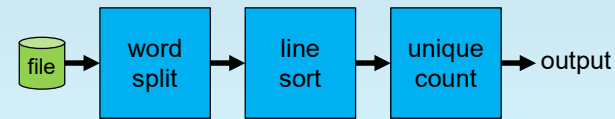Reset FSM to "no word yet" state.

29

Already Made Some Decisions for Line Sort

Now on to line sorting…

Remember these decisions:
- a **word** is no more than **500 characters**,
- no more than **5,000 lines/words,** and
- use **insertion sort**,

file → word split → line sort → unique count → output

30

Line Sort Consists of Three Parts: Read, Sort, Print

Line sort **decomposes into three parts**:
- **read lines** from **stdin** into an array
- **sort lines** using insertion sort
- **print lines** to **stdout**.

Line sort's **main** simply
- checks the command line invocation, then
- calls three functions
- corresponding to the tasks above.

31

Line Sort Uses Array of **char*** with Dynamic Strings

Line sort **reads lines**
- **using fgets** from **stdin** into a buffer, then
- dynamically **allocates a copy** with **strdup**
- and **stores** the new pointer
- **into an array of char*.**

Line sort does not **free** the lines.
- Program memory discarded at termination,
- but be careful.

32