Loops

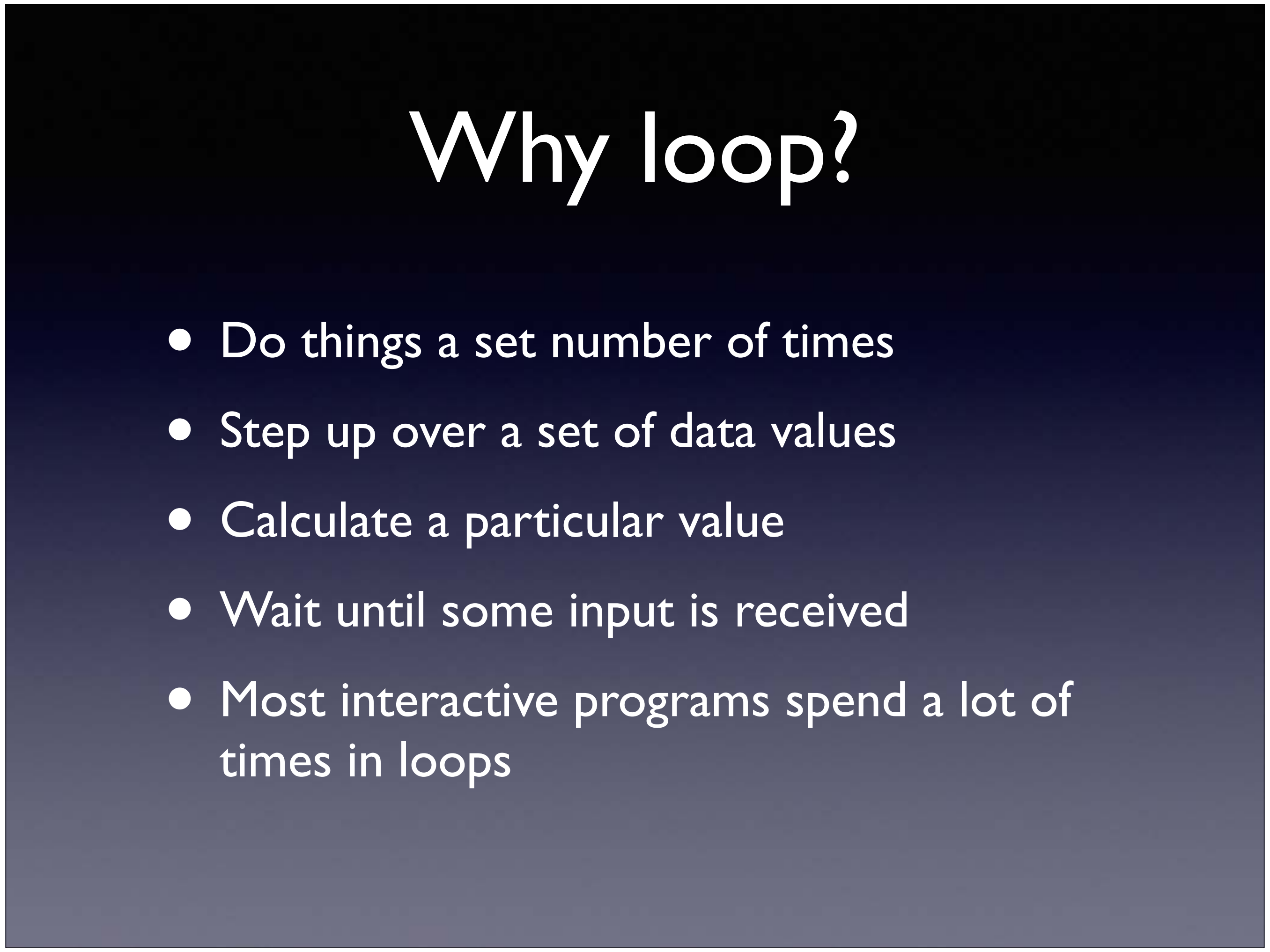
Steven R. Bagley

Program Flow

- Functions allow us to divert into another block of code (possibly several times)
- `if` statements let us decide whether a block of code gets executed or not
- Repeat a block of code several times
- The *loop*

Loops

- Execute a series of statements repeatedly
- Until some condition is met
- C provides three different type of loops
- `while`, `do {...} while`, `for`



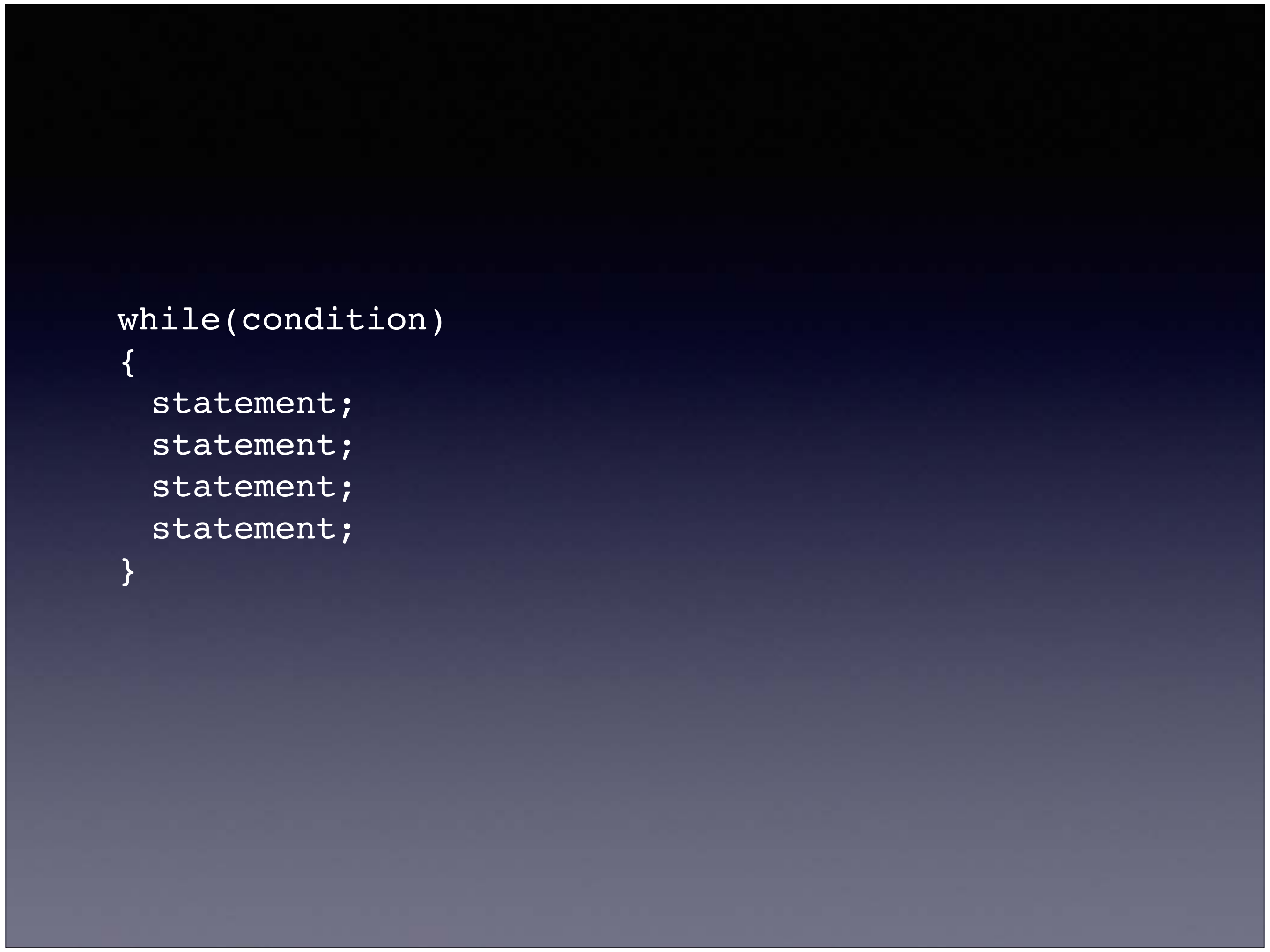
Why loop?

- Do things a set number of times
- Step up over a set of data values
- Calculate a particular value
- Wait until some input is received
- Most interactive programs spend a lot of times in loops

(e.g. using Newton-Raphson to calculate a square root, or the HCF algorithm dave showed you)

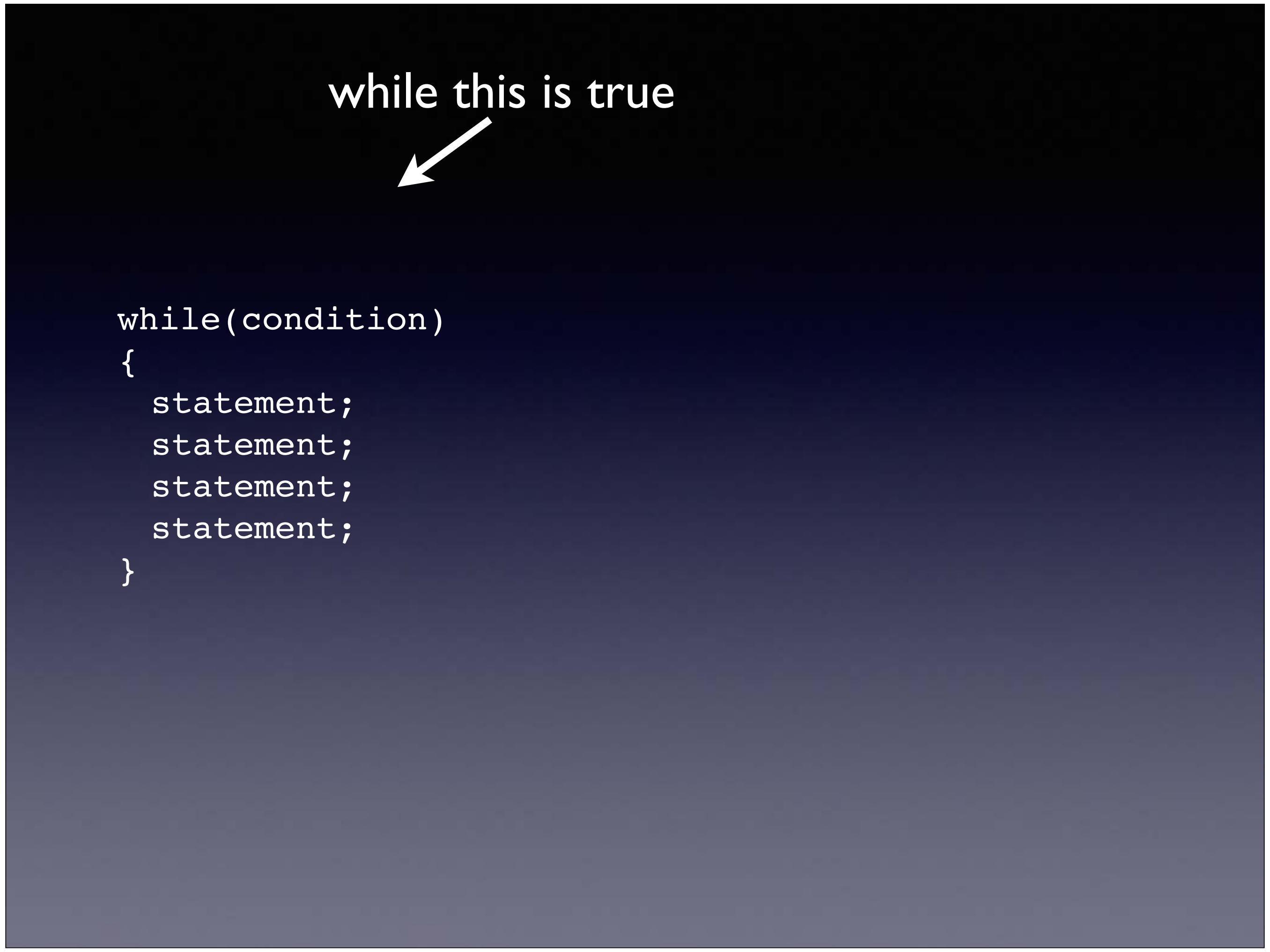
`while` loop

- The `while` loop is probably the simplest
- Has two parts
    - A block of statements to execute
    - A conditional
- While the conditional evaluates to true
- Execute the statements



```
while(condition)
{
  statement;
  statement;
  statement;
  statement;
}
```

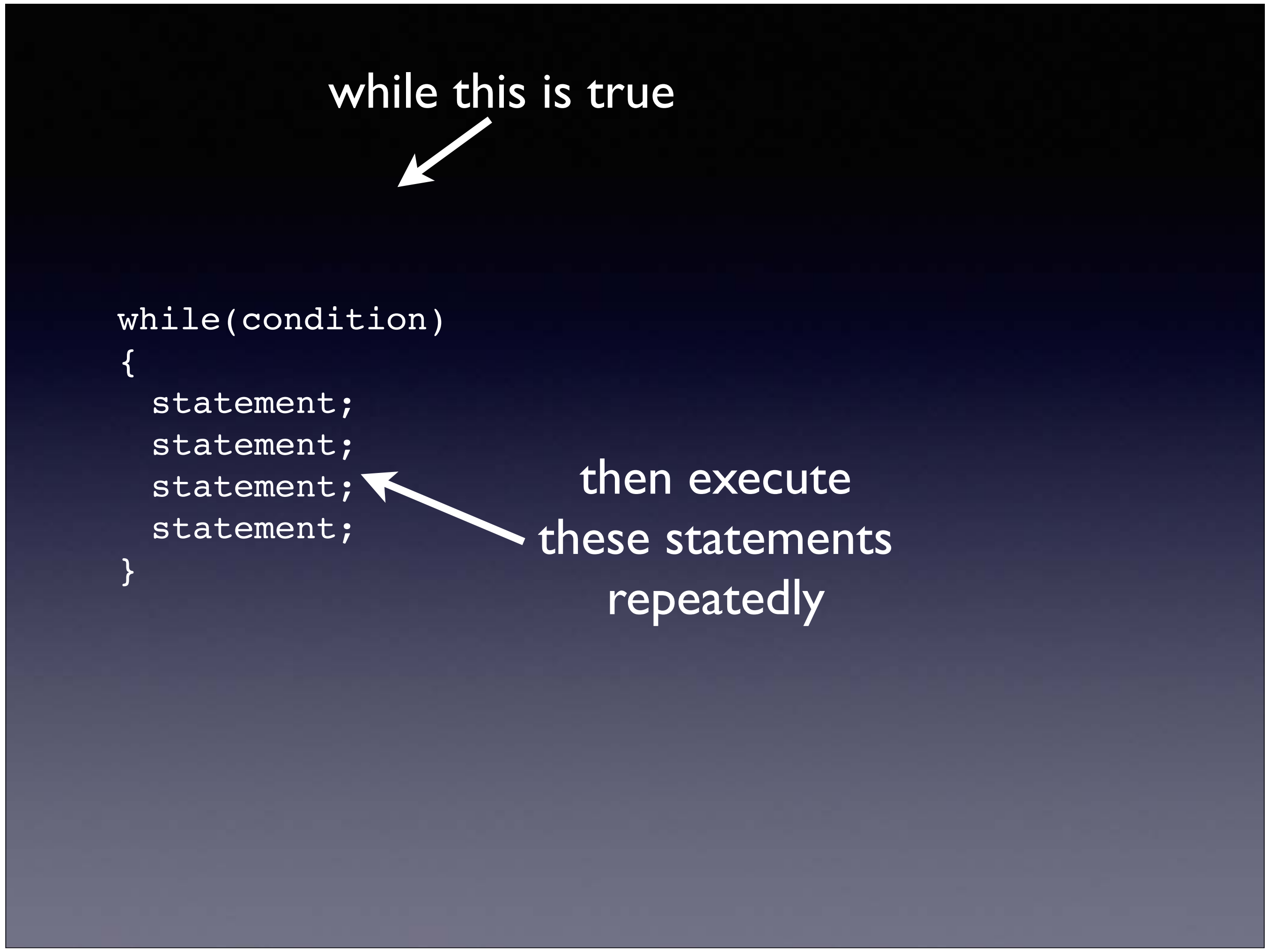
Demo
Does this look familiar?



while this is true

```
while(condition)
{
  statement;
  statement;
  statement;
  statement;
}
```

Demo
Does this look familiar?

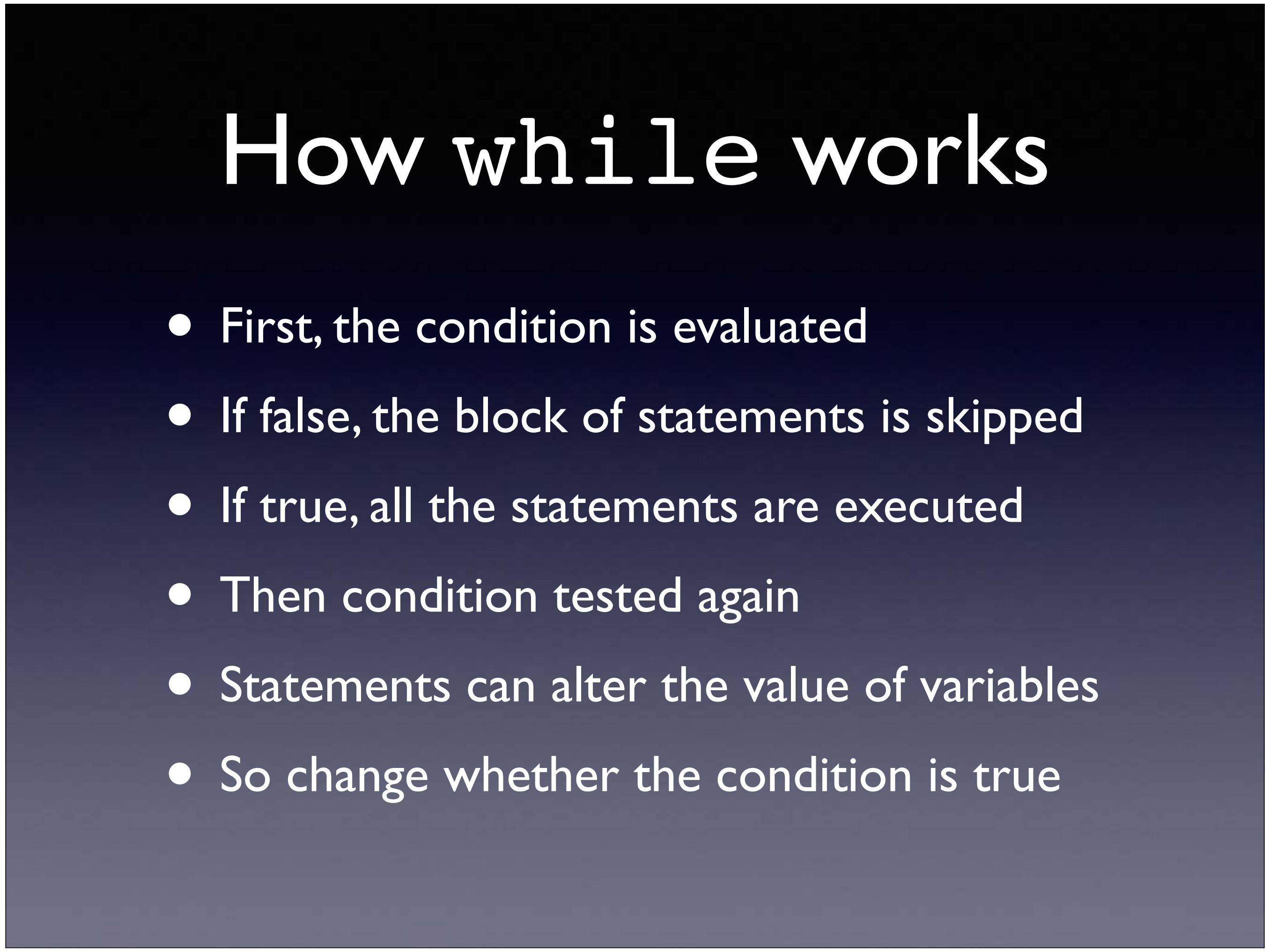


while this is true

```
while(condition)
{
  statement;
  statement;
  statement;
  statement;
}
```

then execute these statements repeatedly

Demo
Does this look familiar?



How `while` works

- First, the condition is evaluated
- If false, the block of statements is skipped
- If true, all the statements are executed
- Then condition tested again
- Statements can alter the value of variables
- So change whether the condition is true

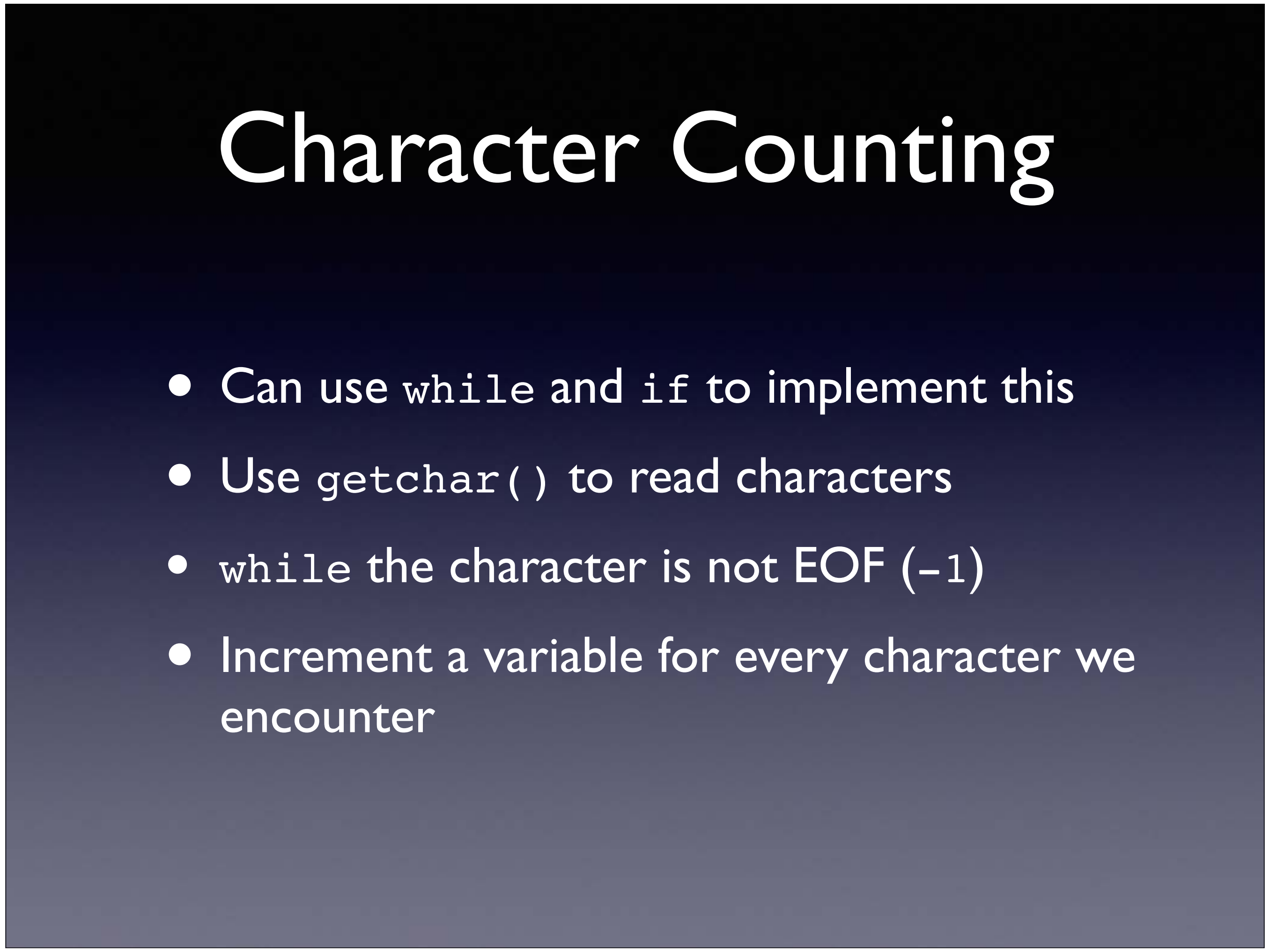
Go demo make a program that calculates a range of temperatures

Interlude

Counting Words

Counting Characters

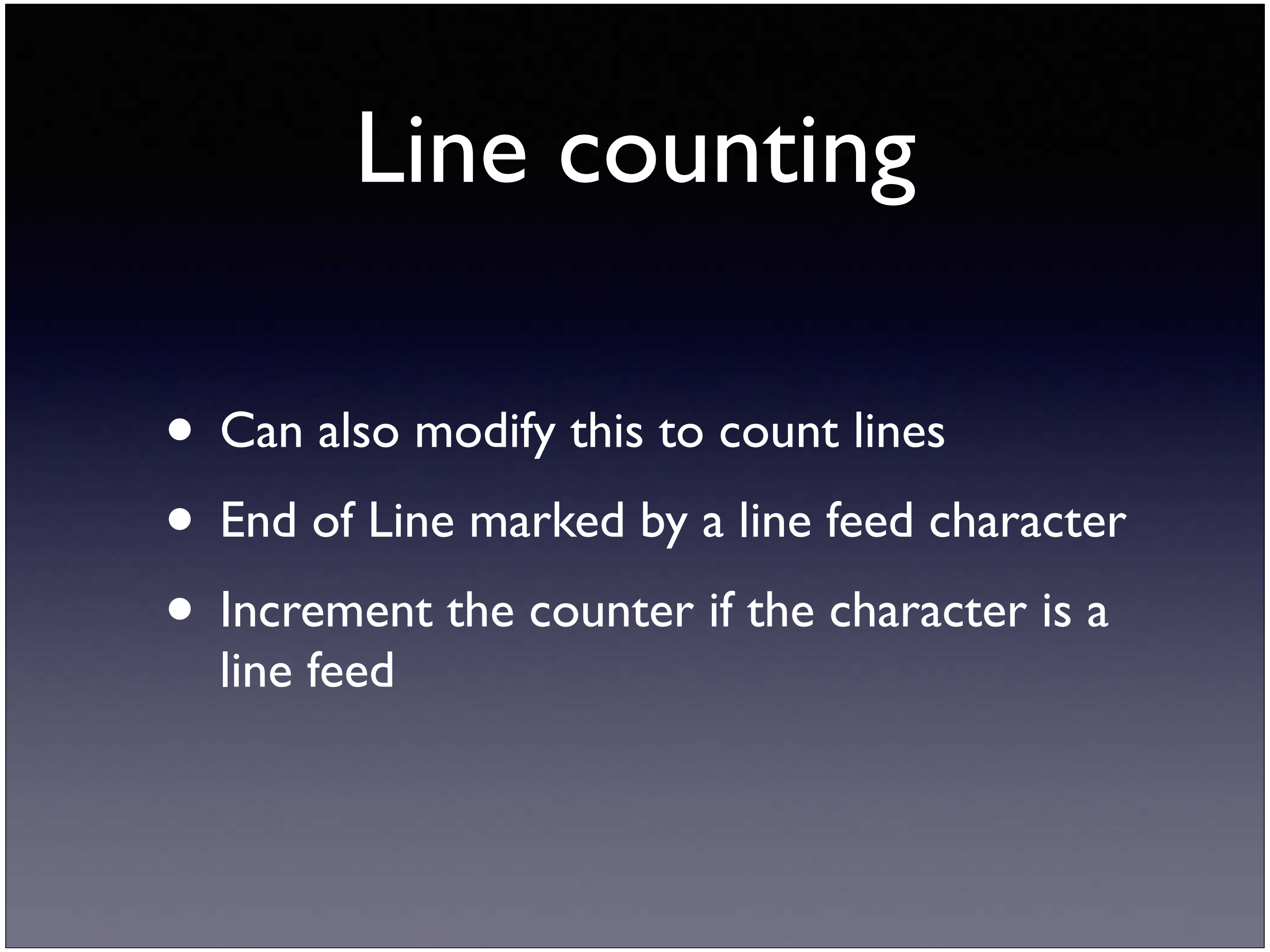
Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, “and what is the use of a book,” thought Alice “without pictures or conversation?”



Character Counting

- Can use `while` and `if` to implement this
- Use `getchar()` to read characters
- `while` the character is not EOF (`-1`)
- Increment a variable for every character we encounter

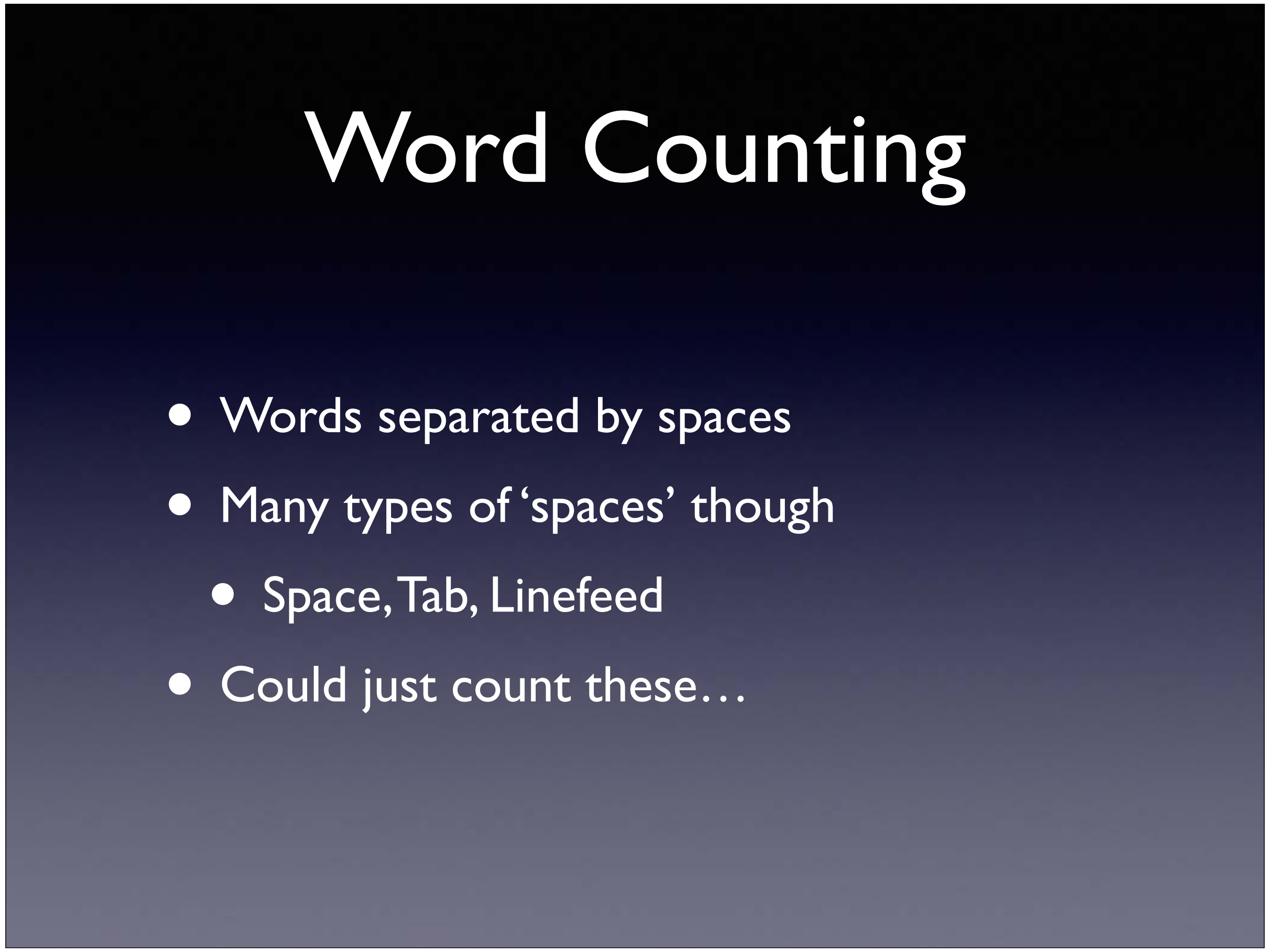
EOF == end of file
Implement



Line counting

- Can also modify this to count lines
- End of Line marked by a line feed character
- Increment the counter if the character is a line feed

Go add this to our program



Word Counting

- Words separated by spaces
- Many types of ‘spaces’ though
  - Space, Tab, Linefeed
- Could just count these…

Go Count...

Word Counting

- Problem…
- Words separated by space, tab or linefeed
- But there may be more than one space
- Therefore, can’t just increment counter when we find one
- That would count spaces, not words…