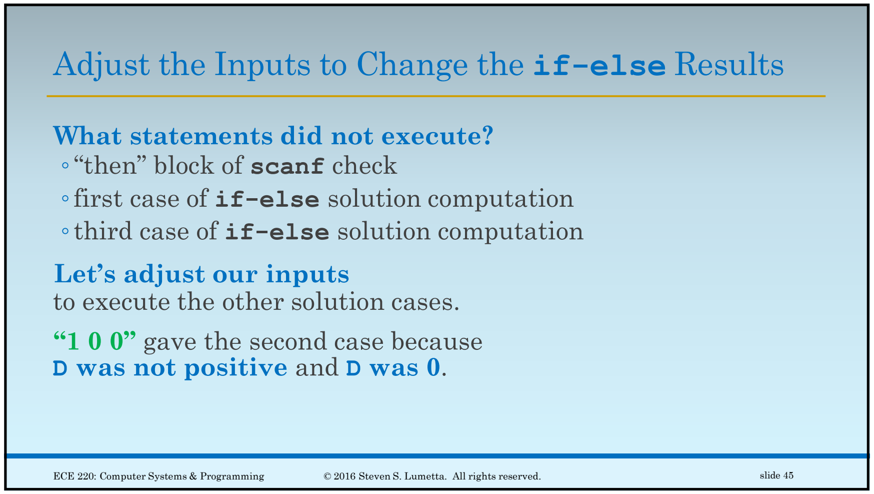


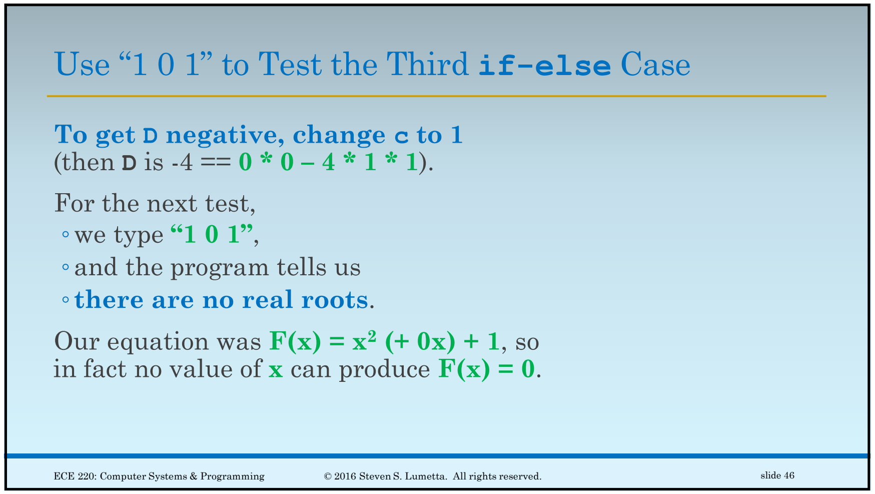
Adjust the Inputs to Change the `if-else` Results

What statements did not execute?
- "then" block of `scanf` check
- first case of `if-else` solution computation
- third case of `if-else` solution computation

Let's adjust our inputs
to execute the other solution cases.

"1 0 0" gave the second case because
`D` was not positive and **`D` was 0**.

45



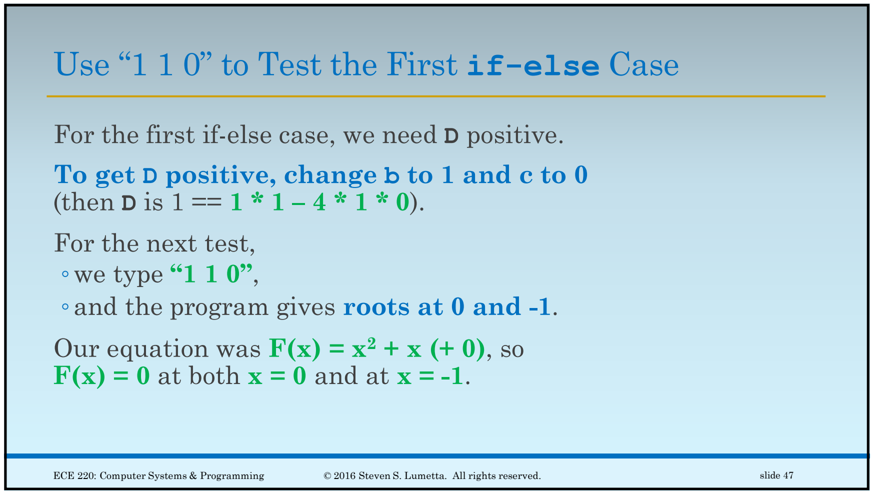
Use "1 0 1" to Test the Third `if-else` Case

To get `D` negative, change `c` to 1
(then `D` is -4 == **0 * 0 – 4 * 1 * 1**).

For the next test,
- we type **"1 0 1"**,
- and the program tells us
- **there are no real roots**.

Our equation was **$F(x) = x^2$ (+ 0x) + 1**, so
in fact no value of **x** can produce **F(x) = 0**.

46



Use "1 1 0" to Test the First `if-else` Case

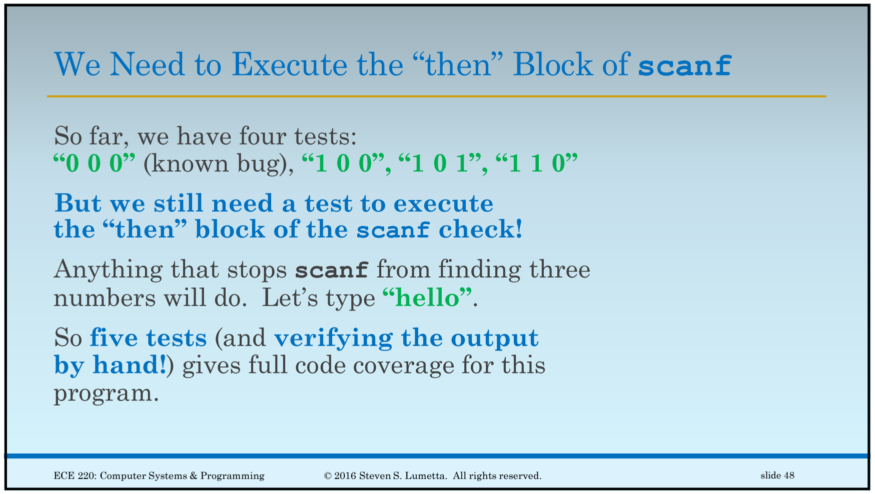
For the first if-else case, we need `D` positive.

To get `D` positive, change `b` to 1 and c to 0
(then `D` is 1 == **1 * 1 – 4 * 1 * 0**).

For the next test,
- we type **"1 1 0"**,
- and the program gives **roots at 0 and -1**.

Our equation was **$F(x) = x^2 + x$ (+ 0)**, so
F(x) = 0 at both **x = 0** and at **x = -1**.

47



We Need to Execute the "then" Block of `scanf`

So far, we have four tests:
"0 0 0" (known bug), **"1 0 0", "1 0 1", "1 1 0"**

**But we still need a test to execute
the "then" block of the `scanf` check!**

Anything that stops `scanf` from finding three numbers will do. Let's type **"hello"**.

So **five tests** (and **verifying the output by hand!**) gives full code coverage for this program.

48