University of Illinois at Urbana-Champaign Dept. of Electrical and Computer Engineering

### ECE 120: Introduction to Computing

Examples of C Programs with Loops

ECE 120: Introduction to Computing

ECE 120: Introduction to Computing

© 2016 Steven S. Lumetta. All rights reserved.

# Time for Some Detailed Examples

Let's do some examples of program execution.

Before you can execute a program, you need to **learn how to compile**.

You will learn that in the lab.

You should also **take a look at the style guidelines** for the class (see the Wiki).

The examples obey most style rules, but space is tight in slides.

You may want to get out a sheet of paper...

ECE 120: Introduction to Computing

© 2016 Steven S. Lumetta. All rights reserved.

# Let's See How This Loop Works

```
/* Print 20 Fibonacci numbers. */
int A = 1; int B = 1; int C; int D;
for (D = 0; 20 > D; D = D + 1) {
    printf ("%d\n", A);
    C = A + B;
    A = B;
    B = C;
}
NOTE: Example programs are available online.
Feel free to try them before/during/after class.
```

© 2016 Steven S. Lumetta. All rights reserved

slide 3

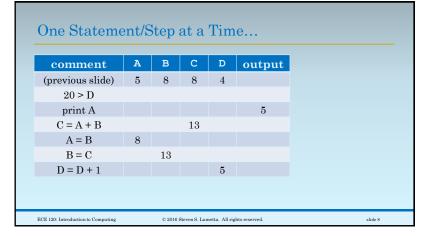
# One Statement/Step at a Time...

before loop			С	D	output
	1	1	bits	bits	
init				0	
20 > D					
print A					1
C = A + B			2		
A = B	1				
B = C		2			
D = D + 1				1	

comment	A	В	С	D	output
previous slide)	1	2	2	1	
20 > D					
print A					1
C = A + B			3		
A = B	2				
B = C		3			
D = D + 1				2	

comment	A	В	С	D	output	
(previous slide)	2	3	3	2		
20 > D						
print A					2	
C = A + B			5			
A = B	3					
B = C		5				
D = D + 1				3		

comment	A	В	С	D	output	
(previous slide)	3	5	5	3		
20 > D						
print A					3	
C = A + B			8			
A = B	5					
B = C		8				
D = D + 1				4		



comment	A	В	С	D	output	
(previous slide)	8	13	13	5		
20 > D						
print A					8	
C = A + B			21			
A = B	13					
B = C		21				
D = D + 1				6		

Each	Loop Iteration Prints One Number	
	tput column on the last few slides	
	ces the first twenty numbers in the	
	acci sequence (on separate lines, without	
comma	ıs):	
	$1, 1, 2, 3, 5, 8, 13, \dots, 6765$	

© 2016 Steven S. Lumetta. All rights reserved.

ECE 120: Introduction to Computing

# Remember factorials? N! = N × (N - 1) × ... × 1 The next program... prints a welcome message, asks user to enter a number, uses scanf to get the number, checks that the user typed something valid, calculates the factorial of the user's number, and prints the factorial.

# Recall that main is a Sequence of Statements When we develop a program, • we break down the problem into smaller steps,\* • and express each step with C statements. The six steps on the previous slide • Are written using C statements • And appear in order in main. \* Part 4 of our class describes a systematic way to do so. Also see P&P Ch. 6.

### Before Statements, We Declare Variables

We need two variables.

- In practice, a programmer may decide to declare more variables as they write statements.
- This program is already finished, so we know how many variables it needs...

```
int number;
/* number given by user  */
int factorial;
/* factorial of user's number */
```

ECE 120: Introduction to Computing

© 2016 Steven S. Lumetta. All rights reserved

. .

slide 15

# How are Variable Names Chosen?

```
int number;

/* number given by user */

int factorial;

/* factorial of user's number */

Variable names

• are chosen to describe their meaning,

• but we use comments to give further details.

These variable names are all lower-case.

Be consistent in how you use case with variable names in a program.
```

ECE 120: Introduction to Computing

© 2016 Steven S. Lumetta. All rights reserved.

# Use printf to Write to the Display

ECE 120: Introduction to Computing © 2016 Steven S. Lumetta. All rights reserved

## Next Step: Wait for the User to Type a Number

After asking the user to enter a number,

- $^{\circ}\, the\; program\; waits\; for\; the\; user\;$
- to type a decimal value using scanf.

scanf ("%d", &number)

The format specifier %d tells scanf to convert decimal ASCII to 2's complement.

The expression &number tells scanf to store the result into the variable number

ECE 120: Introduction to Computing

© 2016 Steven S. Lumetta. All rights reserved.

slide 16

### Always Check the Return Value!

```
Remember that scanf also
• returns 1 if successful (# of conversions)
• returns -1 if the user typed something that isn't a decimal number (such as "hahahaha" ... those humans!)

A program can use the return value (the value of the scanf expression) to determine
```

A program can use the return value (the value of the scanf expression) to determine what has happened...

ECE 120: Introduction to Computing

© 2016 Steven S. Lumetta. All rights reserved

1:1 15

### Next Step: Quit if the User Doesn't Behave

```
if (1 != scanf ("%d", &number)) {
    printf ("Only integers, please.\n");
    return 3; /* Program failed. */
}
The program uses an if statement
to check the result of scanf.

If the user doesn't type a number, the program...
    prints an error message, then
    terminates and tells the OS that something
    went wrong (non-zero by convention).
```

© 2016 Steven S. Lumetta. All rights reserved

### Time for Some Real Work!

• to make the code more readable.

ECE 120: Introduction to Computing

• and in expressions

© 2016 Steven S. Lumetta. All rights reserved.

slide 19

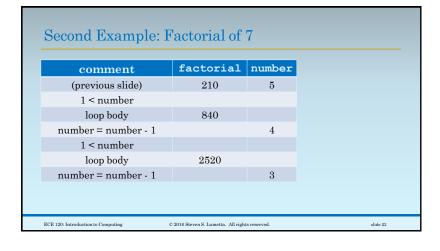
### Example: Factorial of 4

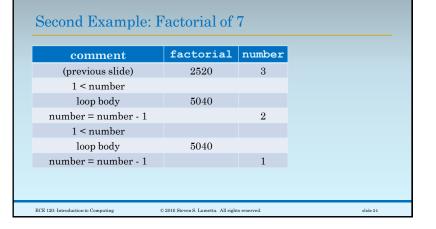
ECE 120: Introduction to Computing

comment	factorial	number
before loop	bits	4
init	4	
1 < number		
loop body	12	
number = number - 1		3
1 < number		
loop body	24	
number = number - 1		2
ECE 120: Introduction to Computing	© 2016 Steven S. Lumetta. All right	s reserved.

comment	factorial	number	
(previous slide)	24	2	
1 < number			
loop body	24		
number = number - 1		1	
1 < number			
after loop	24	1	

	Factorial of		
comment	factorial	number	
before loop	bits	7	
init	7		
1 < number			
loop body	42		
number = number - 1		6	
1 < number			
loop body	210		
number = number - 1		5	





#