The "One Hot" Method for FSM

Ethics in CS ; Impact of CS

CS 64: Computer Organization and Design Logic
Lecture #16
Fall 2018

Ziad Matni, Ph.D.
Dept. of Computer Science, UCSB

Administrative

- Lab #9 Due Friday
- Lab #10 (issued today) Due Friday

- Practice Exam is online

- Your final exam is on

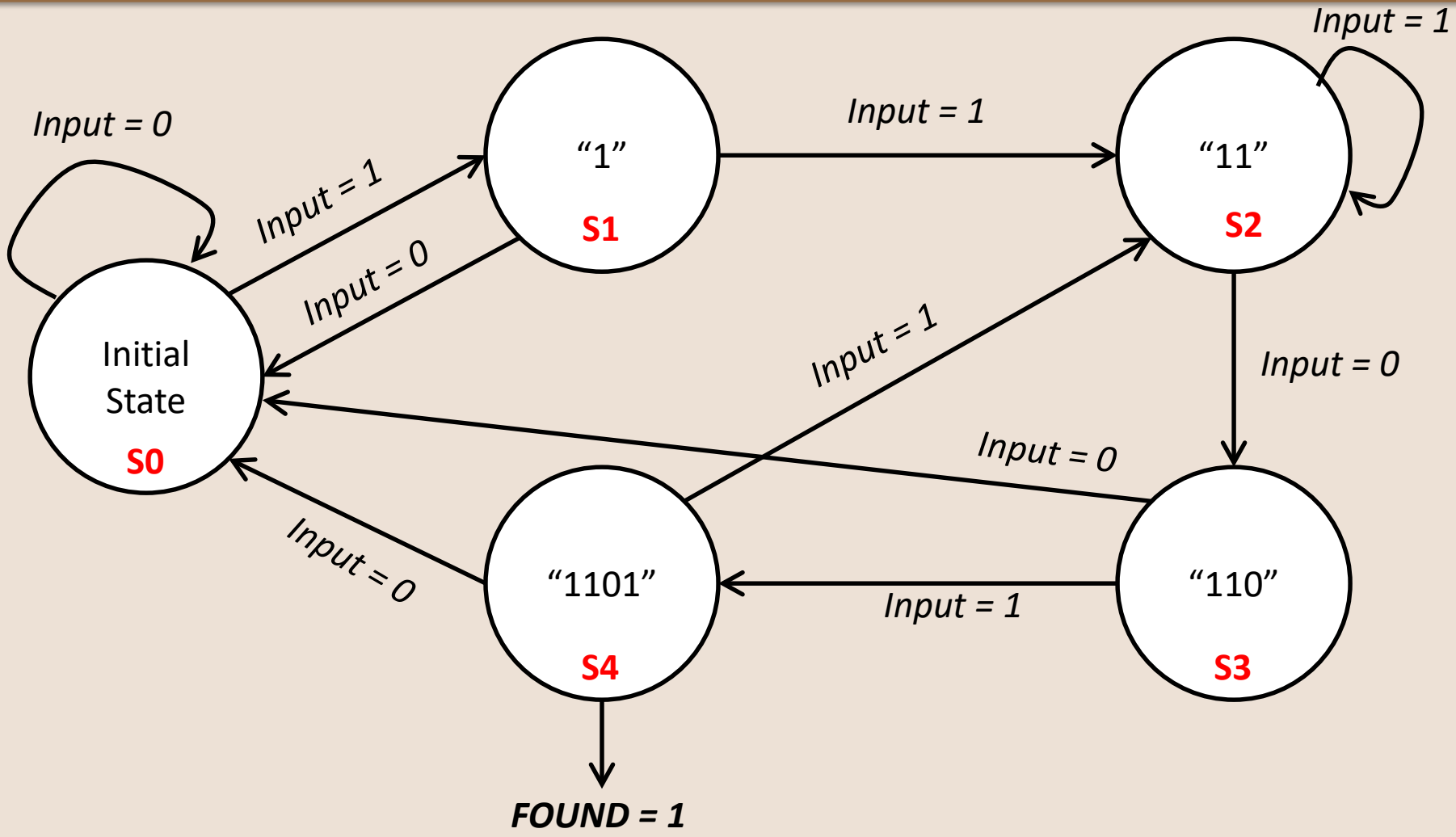
  Wednesday, Dec. 12th at 8:00 AM

  – In this classroom
  – Be ON TIME!

Lecture Outline

- One-Hot Method for FSM
- Ethical Considerations in CS
- Code of Ethics in the Workplace
- CS's Impact on Society

Find “1101” Pattern FSM



Representing The States

- How many bits do I need to represent all the states in this "Detect 1101" Machine?
- There are 5 unique states (including "init")
  - So, 3 bits
- How many D-FFs should I have to build this machine?
  - 3 bits = 3 D-FFs

State	B2	B1	B0
Initial	0	0	0
Found "1"	0	0	1
Found "11"	0	1	0
Found "110"	0	1	1
Found "1101"	1	0	0

Designing the Circuit for the FSM

1. We start with a T.T
   - Also called a “State Transition Table”
2. Make K-Maps and simplify
   - Usually give your answer as a “sum-of-products” form
3. Design the circuit
   - Have to use D-FFs to represent the state bits

1. The Truth Table (The State Transition Table)

	CURRENT STATE			INPUT(S)	NEXT STATE			OUTPUT(S)
State	**B2**	**B1**	**B0**	**I**	**B2***	**B1***	**B0***	**FOUND**
Initial	0	0	0	0	0	0	0	0
				1	0	0	1	0
Found “1”	0	0	1	0	0	0	0	0
				1	0	1	0	0
Found “11”	0	1	0	0	0	1	1	0
				1	0	1	0	0
Found “110”	0	1	1	0	0	0	0	0
				1	1	0	0	0
Found “1101”	1	0	0	0	0	0	0	1
				1	0	1	0	1

2. K-Maps for B2* and B1*

- You need to do this for **all** 3 state outputs
- `B2* = !B2.B1.B0.I`
  - No further simplification

B2.B1 B0.I	00	01	11	10
00				
01				
11		**1**		
10				

- `B1* = !B2.!B1.B0.I`
  `+ B2.!B1.!B0.I`
  `+ !B2.B1.!B0`

B2.B1 B0.I	00	01	11	10
00		**1**		
01		**1**		**1**
11	**1**			
10				

2. K-Map for B0*
Output FOUND

- B0* = !B2.!B1.!B0.I
  + !B2.B1.!B0.!I

B2.B1 B0.I	00	01	11	10
00		1		
01	1			
11				
10				

- FOUND = B2.!B1.!B0
  - Note that FOUND does not need a K-Map. It is always "1" (i.e. True) when we are in state S4 (i.e. when B2=1, B1=0, B0=0)

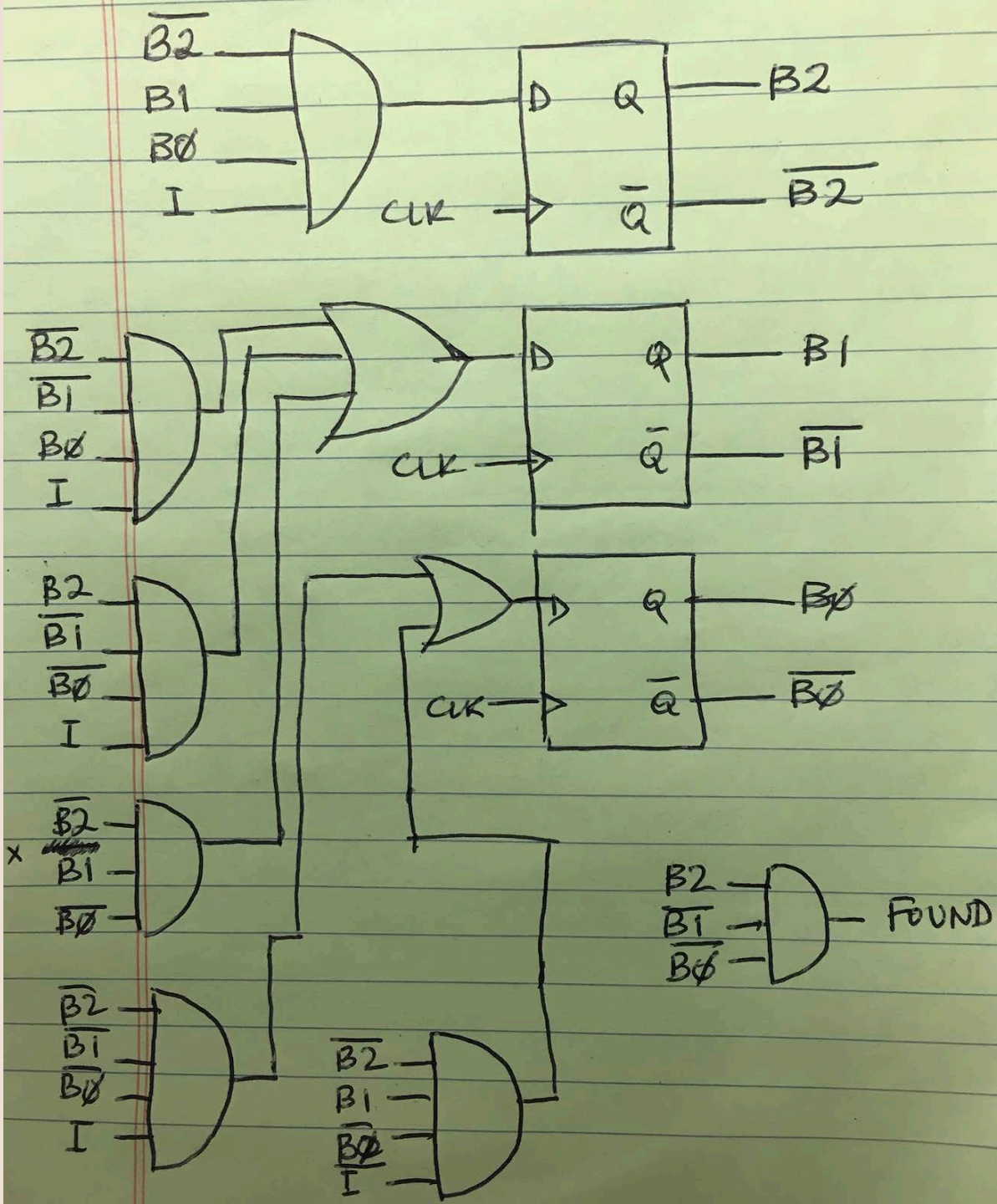
3. Design the Circuit

Note that CLK is the input to ALL the D-FFs' clock inputs. This is a ***synchronous machine***.

Note the use of labels (example: B2 or B0-bar) instead of routing wires all over the place!

Note that I issued both B*n* and B*n*-bar from all the D-FFs – it makes it easier with the labeling and you won't have to use NOT gates!

Note that the sole output (FOUND) does **not** need a D-FF because it is **NOT A STATE BIT!**



The “One Hot” Method

- Most popularly used in building FSMs
- Give each state it’s own D-FF output
  - **# of FFs needed = # of states**
  - You end up using MORE D-FFs, but the implementation is easier to automate
- Inputs to the D-FFs are combinatorial logic that can simplified into a “sum-of-products” type of Boolean expression
  - No need to go through T.T.s and K-Maps
- Current CAD software can do this automatically
- Implemented with FPGA integrated circuits (“chips”)

Encoding our States

Per the last example ("1101 Detector"): We had 5 separate states, so we're going to need 5 bits (i.e. 5 DFFs) to describe the states:

NAME		"Regular" Code	"One Hot" Code	OUTPUTS
Initial State	S0	000	00001	
"1"	S1	001	00010	
"11"	S2	010	00100	
"110"	S3	011	01000	
"1101"	S4	100	10000	FOUND

- Advantage of this "One Hot" approach?
  - When we implement the machine with circuits, we can use a D-FF for every state (so, in this example, we'd use 5 of them)

Using the "One Hot" Code to Determine the Circuit Design

- Every state has 1 D-FF
- We can see that (follow the arrows!!):

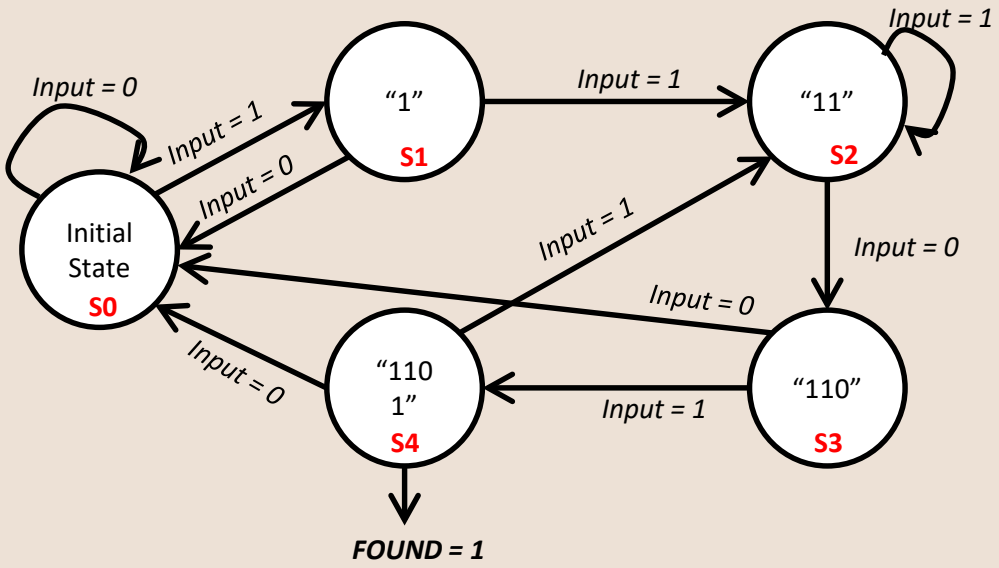
S0* = S0.$\overline{I}$ + S1.$\overline{I}$ + S3.$\overline{I}$ + S4.$\overline{I}$

S1* = S0.I

S2* = S1.I + S2.I + S4.I

S3* = S2.$\overline{I}$

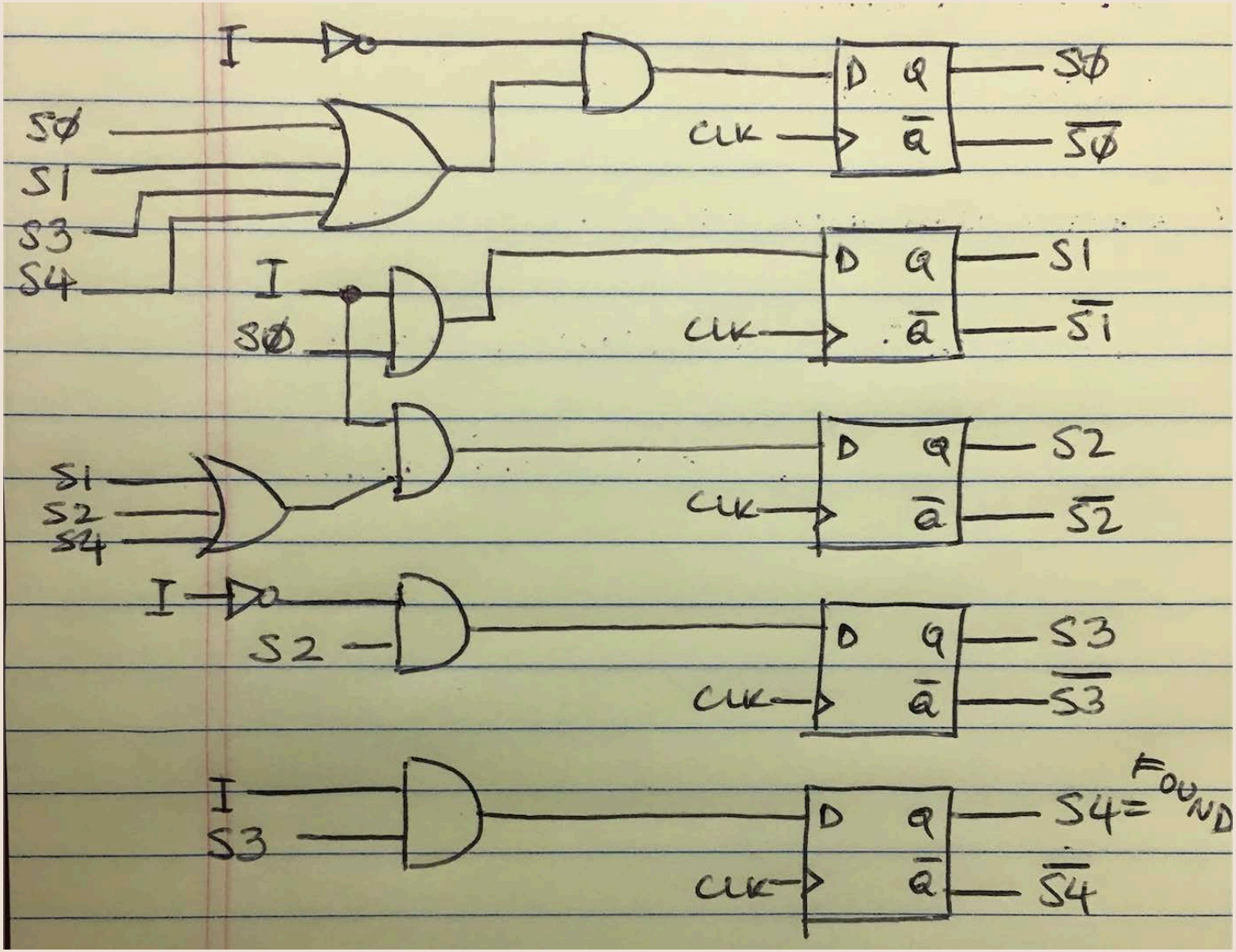
S4* = S3.I



Also, when S4 is True, FOUND is True, i.e. **FOUND = S4**

We have now described ALL the outputs of the machine as combinations of certain inputs WITHOUT needing to do T.T. & K-Maps!

Implementing the Circuit For "Detect 1101" FSM Using the "One Hot" Method



Ethics

- Moral principles that govern a person's behavior
- Attempts to answer questions like:
  "What is the best way people to do something?"
  "What actions are right or wrong"
- In CS, it's not just about the obvious questions, like:
  "Is it ok to copy someone else's code and use it?" *NO*
  "Can I take this mouse pad from work?" *NO*
  "I mean, come ooooon, it's just a mouse pad..." *NO!!!*

Ethical Considerations in CS

- Our work in CS affects people (why do it otherwise?!)
  - Ourselves
  - Our work colleagues
  - Our professional community
  - Society at large

Ethics in CS notes the following:

- Our activities and choices affect other people in significant ways
- We have principles and guidelines that guide ethical action

Ethical Considerations in CS

- Act consistently with the **public interest**, your **clients**, your **employer**, your **colleagues**.
- Make your products/services meet the **highest professional standards** possible.
- Maintain **integrity** in your work. Maintain a good reputation for yourself and your profession
- If you're a manager, **promote an ethical approach** to your work and your team's work.
- Keep **bettering yourself** through education

Who Cares if you Aren't Ethical?

- Everyone does – it's a ***"social contract"***

- If you are not ethical, ***at best the following can happen***…
  - … your job will be at risk
  - … your relationship with others will be at risk
  - … you are likely to be negatively labeled as "unethical" in your professional circle
  - … you will give "a bad name" to yourself, your company/employer, and to the field of CS in general

- ***At worst***, you will have **major** financial/legal ramifications
  - Get fired (and possibly blacklisted)
  - Get sued
  - Get arrested

Professional Guidelines

- The IEEECS/ACM Joint Task Force on Software Engineering Ethics and Professional Practices

"Code of Ethics and Professional Practice"

Purpose:

- Documents the ethical and professional obligations of software engineers.
- Instructs us about the standards society expects CS professionals to meet.
- What to expect of one another.

Code of Ethics and Professional Practice

Lab 10 – Task 1

- Read the IEEE Computer Society's article
- Then read a collection of case studies on ethics
  - Both in the lab description

- Afterwards, go to an online form.
  You will choose which *clauses* from the code of ethics are more relevant to each case study.
  - Link in the lab description

The Impact of CS in the World

- What do YOU think Computer Science's impact in the world today is?...

The Impact of CS in the World

- Today – more than ever before – CS enables us to make tools that help people:
  - Connect
  - Visualize information
  - Understand the impacts of
    environmental, economic, energy happenings
  - Collaborate and work together

Google Talk at CSIT Conference

Lab 10 – Task 2

- View video of Megan Smith's talk at the 2010 Computer Science & Information Technology (CSIT) Conference about the Impact of CS Worldwide
  - Smith was a VP at Google and then the "U.S. CTO" and Assistant to President Obama
  - Link is in the lab description

- Afterwards, go to an online form.
  You will identify the impact of CS in a variety of areas.
  - Link is… ahhh… you know where…

Your To Dos

- Lab #9 and #10 due end of day Friday!
- Study for your final exams!

- Take a breather and **get enough sleep!**

</LECTURE>