Impact of CS Ethics in CS Prep for Final Exam

CS 64: Computer Organization and Design Logic Lecture #17

Ziad Matni
Dept. of Computer Science, UCSB
Administrative

• Labs 9 & 10 due on Thursday 3/16 (last day of class)

• Midterm #2 grading done
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?”
  “Can I take this mouse pad from work home?”
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 client, your employer, your colleagues.
• Make your products 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/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 ...
  – ... 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 CS in general

• At worst, you 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.
Lab 10 – Task 1

• Read the IEEE Computer Society’s article

• Then read a collection of case studies on ethics.

• Afterwards, go to an online form. You will which clauses from the code of ethics are more relevant to each case study.
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

• Afterwards, go to an online form. You will identify the impact of CS in a variety of areas.
Midterm #2 Results

CS64 MIDTERM #2
Mean = 75.5, Median = 77.5, SD = 12.9
Review Parts of Midterm #2
int main () {
    int arr[5] = {45, 52, 11, 0, 26};
    int size = 5;
    int t0 = arr[0];
    // If a new maximum is found, put it in t0
    for( int t1 = 1; t1 < size; t1++ )
        if(arr[t1] > t0) { t0 = arr[t1]; }
    print(t0);
}

3/13/17
int chowder(int a, int b) {
    int c = 1;
    if (a == b) c = 0;
    else c += chowder(a - 1, b);
    return c;
}

int main () {
    int x = 6, y = 3;
    print (chowder(x, y));
}

.text
main:
    li $a0, 6
    li $a1, 3
    li $v0, 0
    jal chowder
    li $v0, 1
    syscall
    li $v0, 10
    syscall

chowder:
    addiu $sp, $sp, -8  # Prepare to push into the stack
    sw $ra, 0($sp)      # Push $ra into the stack
    sw $s0, 4($sp)      # Push $s0 into the stack
    li $s0, 1           # $s0 is var c and c = 1
    bne $a0, $a1, else  # if a != b, go to “else”
    li $s0, 0           # if a == b, make c = 0
    j return            # go and return c
else:
    addi $a0, $a0, -1   # if a != b, decrement a
    jal chowder         # recursive call to chowder()
    add $s0, $s0, $v0    # c += old value of c
return:
    move $v0, $s0       # Place c into $a0 (to print)
    lw $s0, 4($sp)      # Pop $s0 from the stack
    lw $ra, 0($sp)      # Pop $ra from the stack
    addiu $sp, $sp, 8   # Reset the stack pointer
    jr $ra              # jump back to the initial call
Q7

7. Simplify the following logic expression algebraically.

a) (2 pts) \( X = A.B.C + \overline{A}.\overline{B}.\overline{C} + \overline{A}.\overline{C} + A.B \)

\[
X = A.B(C + 1) + !A.!C.(!B + 1) \\
= A.B + !A.!C
\]

b) (2 pts) \( F = X.Y + Y.(\overline{X} + X.Z) + X \)

\[
F = X.Y + !X.Y + X.Y.Z + X \\
= Y. (X + !X + X.Z) + X \\
= Y. (1 + X.Z) + X \\
= Y + X
\]

c) (3 pts) \( G = (A \text{ XOR } B) + A.B \)

You have to end up with a single OR operation for full credit!

\[
G = A.!B + !A.B + A.B \\
= A.(!B + B) + !A.B \\
= A + !A.B \\
= A + B
\]
If $F = A.B.C + \bar{B}.\bar{C}$, then express $\bar{F}$ as algebraically simplified as you can as a sum-of-products.

\[
\bar{F} = (A.B.C). (\bar{B}.\bar{C}) = (\bar{A} + \bar{B} + \bar{C}).(B + C) = \bar{A}.B + B.B + \bar{C}.B + \bar{A}.C + \bar{B}.C + \bar{C}.C = \bar{A}.B + \bar{A}.C + \bar{B}.C + B.\bar{C}
\]
Q10

- First, note that all muxes have the function: \( Y = I_0 \cdot \bar{S} + I_1 \cdot S \)
- And in this case: \( I_0 = A \cdot B, \quad I_1 = B \), so you see that:
  \[ Y = A \cdot B \cdot \bar{S} + B \cdot S \]
- The truth table would look like this:

<table>
<thead>
<tr>
<th>S</th>
<th>A</th>
<th>B</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Note:
- When \( S = 0 \), \( Y = A \cdot B \)
- When \( S = 1 \), \( Y = B \)
Q11

- K-Map

- Optimized sum-of-products

\[
F = \overline{X}.Z + W.Z + \overline{W}.X.Y.\overline{Z}
\]
</LECTURE>