

University of California, Berkeley – College of Engineering

Department of Electrical Engineering and Computer Sciences

Spring 2012 Instructor: Dan Garcia 2012-03-22

CS10 PAPER MIDTERM

<i>Last Name</i>	
<i>First Name</i>	
<i>Student ID Number</i>	
<i>cs10- Login First Letter</i>	a b c d e f g h i j k l m
<i>cs10- Login Last Letter</i>	a b c d e f g h i j k l m n o p q r s t u v w x y z
<i>The name of your LAB TA (please circle)</i>	Aijia Luke Navin Pierce Rabbit Samir
<i>Name of the person to your Left</i>	
<i>Name of the person to your Right</i>	
<i>All my work is my own. I had no prior knowledge of the exam contents nor will I share the contents with others in CS10 who have not taken it yet. (please sign)</i>	

Instructions

- Don't Panic!
- This booklet contains 6 pages including this cover page. Put all answers on these pages; don't hand in any stray pieces of paper.
- Please turn off all pagers, cell phones and beepers. Remove all hats and headphones.
- You have 110 minutes to complete this exam. The midterm is closed book, no computers, no PDAs, no cell phones, no calculators, but you are allowed two double-sided sets of notes. There may be partial credit for incomplete answers; write as much of the solution as you can. When we provide a blank, please fit your answer within the space provided.

Question	1	2	3	4	5	6	7	8	9	10	11	12	13	14	Online	Total
Points	2	2	2	2	2	2	2	2	2	2	6	4	10	10	10	60
Score																

If you can draw and you have time, feel free to doodle all over this front page!

Questions with one sentence answers (2 pts each)

Question 1: One big idea in this course is that often technology has unintended *negative consequences*. What was one of the negative consequences of the advent of geo-tagged and date-stamped YouTube videos?

Question 2: Amdahl's law states that if a program contains a $1/5$ serial (non-parallel) portion and $4/5$ parallel portion is run on a fast machine with a near-infinite number of cores, what's the *most speedup you could ever achieve* over a machine with a single core?

Question 3: Judah Schwartz said there was a continuum of computer uses in education: from *tools* through *microworlds* through *courseware*. BYOB would be a tool, but what would a microworld for teaching graphically recursive structures in BYOB look like?

Question 4: One of the most important aspects of HCI is the *Iterative Design Cycle*. What is that?

Question 5: What is the "*free performance lunch*" that's now over?

Question 6: Computer performance is continuing to increase exponentially. This means one by one we'll be able to *strongly solve all the world's board games*. True or False? Explain.

Question 7: From an engineering perspective, Twitter doesn't want everyone with an account to "follow" everyone else with an account. Why would the financial officers Twitter not want that either? (*Hint: think about what Rushkoff said in his "Program or be Programmed" video – what's really behind Twitter's financial model?*)

Question 8: What is one reason (also provide an example) most of us have *given away* our privacy?

Question 9: If the US Government wanted to perform mind control, what could they require Google to do?

Question 10: Recently, there was "de-escalation" in the age-old copyright war; describe it.

Question 11: *Beethoven wasn't the only great composer...* (6 pts)

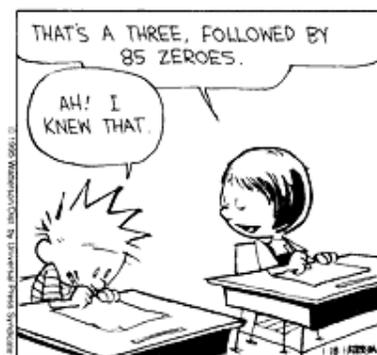
We've provided some helper reporter blocks that work on *both* words and sentences.

Block	Description	Word example	Sentence example
Length	Report the number of letters in a word / words in a sentence		
Unend	Remove the ends of a word / sentence.		
Double	Double a word / sentence		
RightDup	Duplicate the rightmost letter / word		
LeftTrim	Remove the first letter / word from the left		

Fill in the blanks below with only calls to the reporter blocks above: Length, Unend, Double, RightDup and LeftTrim so that the expressions evaluate correctly. Use the techniques from "Writing Scratch/BYOB code on paper". E.g., would be written `RightDup(Double(Bears))`. You may use at most 5 reporter blocks in any one blank. The blanks on the right are for matching right parens.

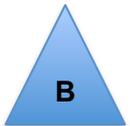
a) _____(I love cal)____ → 9

b) _____(go bears and beat stanford)____ → dan



Question 12: Three body simulation... (4 pts)

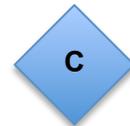
You have three sprites, A, B and C. A is centered on the screen and does nothing (never moves). B and C follow the scripts below, and the pen for C is in the center of the diamond. Draw the pen marks that are made. Assume sprites are allowed to go off the stage.



B

```

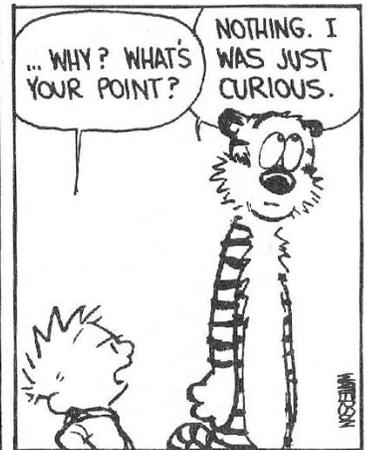
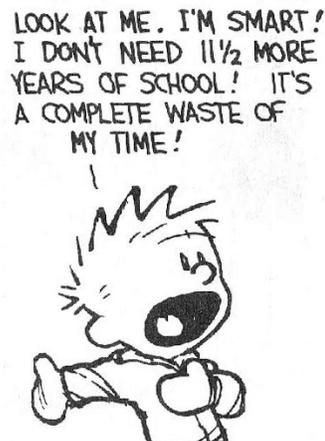
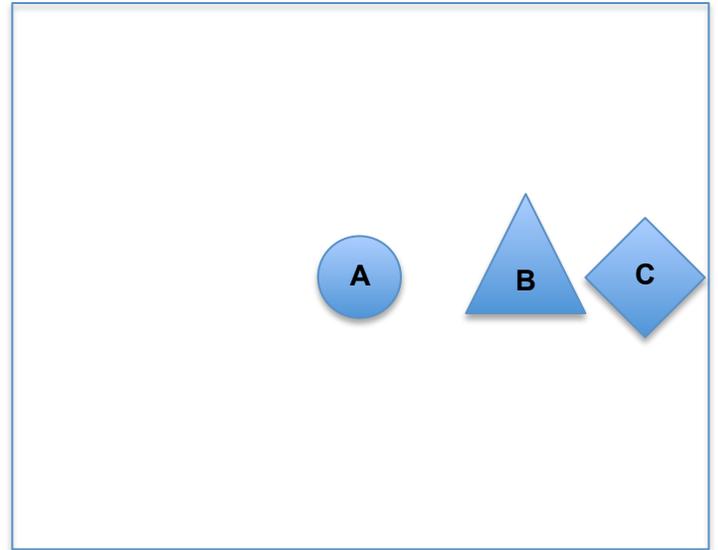
when green flag clicked
  forever loop
    point towards A
    turn 90 degrees
    move 1 steps
        
```



C

```

when green flag clicked
  clear
  pen down
  forever loop
    point towards B
    turn 90 degrees
    move 3 steps
        
```



Question 13: I've got this incredible run going, hold on!! (10 pts)

When a list has many consecutive identical elements in a row, it is called a “run”. We’d like to write a reporter, `longest-run(list)` to return the maximum number of consecutive elements in a list. For example, `longest-run((a b c d d e a b x x x d q))` would return 3 because the highest number of consecutive elements is 3 (the run of 3 x’s in a row). Assume lists have at least one element in them. We’ve tried to write the code ourselves, but unfortunately, it has a bug.

```

longest run list
script variables index longest run current run
set index to 1
set longest run to 1
set current run to 1
repeat until index > length of list
  change index by 1
  if item index - 1 of list = item index of list
    change current run by 1
  if current run > longest run
    set longest run to current run
report longest run
  
```

a) If `length-of(list)` is a linear operation, what is the order of growth of `longest-run` _____?

b) Complete the sentence. The shortest list that triggers the bug is _____.

The `longest-run` function *should* return _____ for this list, but our buggy version returns _____.

c) Describe the smallest fix that fixes the function so that it works as advertised on all non-empty lists.

Question 14: Was this game invented in Italy in the 13th century? (10 pts)

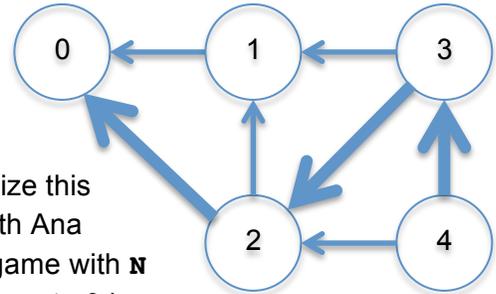
Consider a game similar to the “1,2,...,10” game from lecture. In this game, two players are given a bowl of n candies on the table. They take turns, eating ONE or TWO candies from the bowl. The first player to eat the last candy and leave the bowl empty is the winner. Here’s an example game:

Ana: “Hmm, the number of candies in the bowl is 4. I’ll eat ONE to make it 3.”

Bob: “I’ll eat ONE to make it 2.”

Ana: “I’ll eat TWO to leave an empty bowl. I win!!”

You want to calculate how many different paths (i.e., different games) there could be starting with n candies. For example, with 4 candies, one path was the one Ana and Bob played: “4→3→2→0”. Another might have been “4→2→0”, or “4→3→2→1→0”, or “4→2→1→0”, or “4→3→1→0” (5 paths in total). You decide to visualize this when $n = 4$, and you get the picture on the right. We’ve shown the path Ana and Bob took in bold. You realize the number of different paths in a game with n candies is exactly the same as the number of different ways to go from n to 0 in a visualization like the one shown here.



a) Write a function **Paths** that takes one integer argument n (the starting number of candies), and returns the number of paths in the candy game starting with n candies. For example, **Paths(0)** should return 1 (there’s only one path if you’re already there: “0”). **Paths(1)** should return 1 (one path: “1→0”). **Paths(2)** should return 2 (for the two paths: “2→0” and “2→1→0”), **Paths(4)** should return 5 (as described above), etc.

Paths(N)

```

if ( _____ )
    report ( _____ )
report ( _____ )
    
```

b) What is the order of growth of **Paths(N)**? _____

c) The values returned by **Paths(1)**, **Paths(2)**, **Paths(3)**, etc. form an interesting series. Who was the first person to notice this? _____

