

CS 202: Introduction to Computation
Fall 2010: Exam #1

Name: _____

Question	Possible Points	Received Points
1	20	
2	20	
3	20	
4	20	
5	20	
Total	100	

This exam is closed notes.

You have 50 minutes to complete the 5 questions on this exam.

Please write your answers clearly.

Good luck!

Question 1: Why watch TV when you could be Programming?

- A) A friend of yours has created the Scratch program shown in the appendix. (You may remove it for reference.) The program recommends a TV show based upon how the user answers a series of questions. On the back of the previous page, draw the **decision tree** corresponding to the scripts. Give a descriptive name to each node of the tree, label the transitions between nodes, and show the outputs of the program.

- B) How many different TV shows can this program recommend (across multiple runs of the program)?

- C) In any one run of the program, what is the fewest number of questions the user could be asked (i.e., the best case)? What TV show will be recommended in this case?

- D) In any one run of the program, what is the greatest number of questions the user could be asked (i.e., the worst case)? What TV show will be recommended in this case?

- E) If the user receives the recommendation to watch Lost, what do you know about him or her?

- F) Did your friend create a program that resulted in “good” decision tree? What qualities are good or bad about the resulting decision tree?

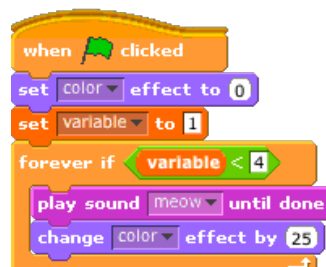
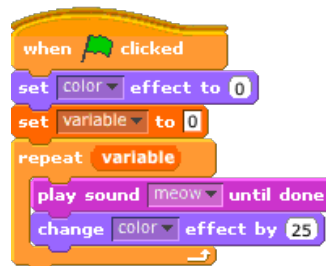
- G) If you were to write a different Scratch program to recommend a TV show, what would be your goal in terms of the number of questions asked? How many questions would it ask in the worst case?

Question 2: Some of these things are not like the others...

Assume you have a cat who is running the following script.



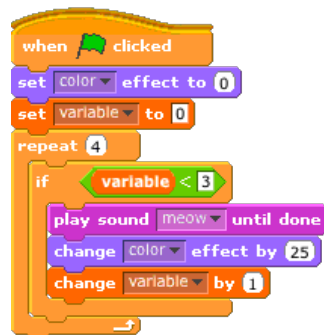
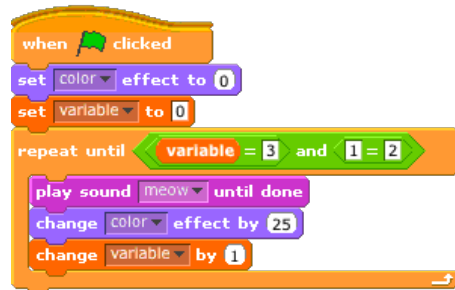
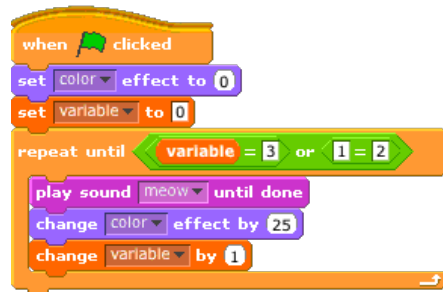
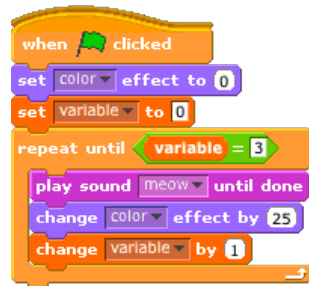
For each of the following scripts activated by “When Green Flag clicked”, say whether or not it results in the same behavior as the original script. If it has different behavior, state **how** the behavior is different and **why**.



Original code (again):

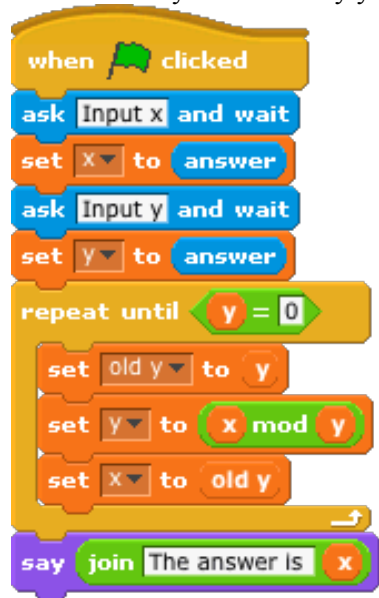


For each of the following scripts, say whether or not it results in the same behavior as the original script. If it has different behavior, state how the behavior is different.



Question 3: How do the variables vary?

Consider the following Script which has access to three variables called x, y, and old y. Remember that the block "x mod y" divides x by y and returns the remainder. For example, $7 \bmod 2 = 1$ and $6 \bmod 2 = 0$.



For the following input values, fill in the table to show the value of each variable at the end of each iteration of the repeat loop. You may not need all of the rows of the table. You may find it useful to show the initial values of each variable in the header row of the table.

A) $x=20, y=5$.

Loop #	old y	y	x
1			
2			
3			
4			

B) $x=21, y=14$.

Loop #	old y	y	x
1			
2			
3			
4			

C) $x=8, y=3$.

Loop #	old y	y	x
1			
2			
3			
4			

Extra Credit (No stress, please): What mathematical function is this script computing? (Hint: It is difficult to figure this out from looking at the code itself; you may find the function easier to uncover by looking at many different inputs and the resulting outputs.)

Question 4: Is there too much going on at one time?

Consider three sets of scripts (A, B, and C). Each set of scripts is a single program and is started when the Green Flag is clicked. Each set of scripts has a single variable: **test**. For each set of scripts, answer these two questions.

1. Which scripts within a set will run **concurrently** with one another (or with part of one another)? Why? You should use the names “script 1” “script 2” and “script 3” to identify each script.
2. What final values could the **test** variable contain after all of the scripts in the set terminate? What different orderings of the scripts could lead to the different values?

A)



B)

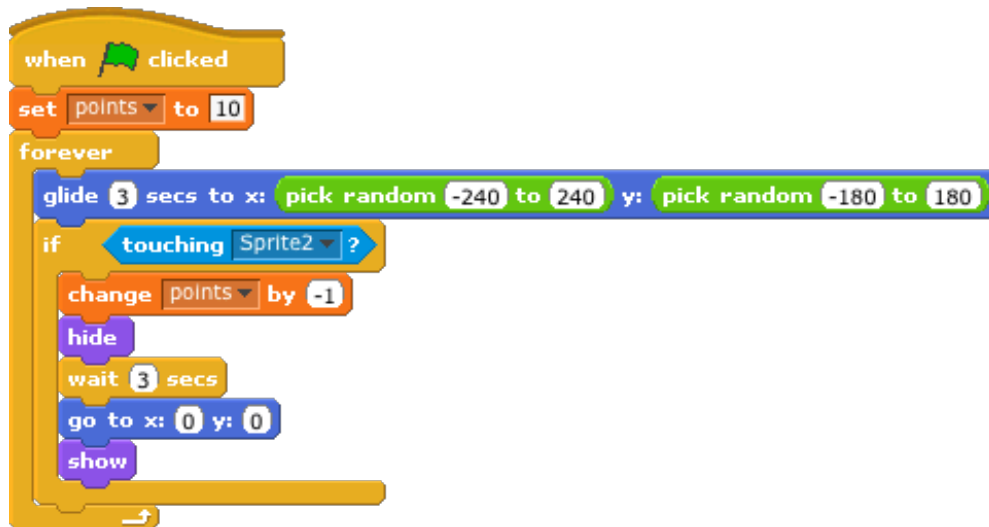


C)



Question 4 Continued (not directly connected to parts A, B, and C)

- D) The following script has a bug in it. The intention was that the Sprite would glide randomly around the stage forever; if it ever touches Sprite2 while it is moving, then the number of points should be decremented and the Sprite should show up at the center of the stage few seconds later. However, when this script is run, it rarely seems to detect that this sprite is touching Sprite2.
- What is the problem?
 - How could you easily fix this program (while still using all of the existing blocks)?



Question 5: Do you want to play a game?

Assume you are playing tic-tac-toe with your friend. You are X and your friend is O. You've each taken 3 turns and the board now looks like this:

X	O	O
O		X
X		

A) To help you make your next move as X, draw the complete **game tree** for the final three moves. Mark whether each final board is a win, lose, or tie for X.

B) Where will you decide to place your next X? How does the game tree guide you to that decision?

Appendix for Question 1. This page may be removed for reference.

