Please read all instructions carefully.

This exam consists of two parts. The first part should be completed without the use of Scratch and your answers should be written on the paper titled "CS39n - Midterm - PART 1". Once you complete Part 1, please write your name on the last page (of part 1) and turn it in before continuing on to Part 2. The second part of the exam will be done on the computer. Please submit part 2 according to the directions in the attached document. The exam and all files are available from the bSpace assignments tab.

This exam is out of 100 points.

Programming Part without Computers:

(7 pts) Prob 1	
(7 pts) Prob 2	
(6 pts) Prob 3	
(8 pts) Prob 4	

Programming Part with Computers:

(16 pts) Prob 5	
(8 pts) Prob 6	

Writing Section:

(16 pts) Prob 7	
(16 pts) Prob 8	
(16 pts) Prob 9	

Total:

Part Without computers

1. This semester, you have encountered two blocks for manipulating boolean variables: **or** and **and**. These two blocks have the truth table as shown. In computer science, there

| and	or	xor
False and False ⇒ False	False or False ⇒ False	False xor False ⇒ False
False and True ⇒ False	False or True ⇒ True	False xor True ⇒ True
True and False ⇒ False	True or False ⇒ True	True xor False ⇒ True
True and True ⇒ True	True or True ⇒ True	True xor True ⇒ False

is another operator that we often use, called **XOR**, also known as exclusive or. This operator is described in the rightmost truth table. In other words, it is an **or** that does not return true if they are both true.

Write XOR two different ways. You may write **TRUE** or **FALSE** in the blanks. You may assume that *a* and *b* are both booleans and may go in the blanks directly if appropriate. You may find the Boolean function **not** useful:

a) using **if/else** but without using **and/or**
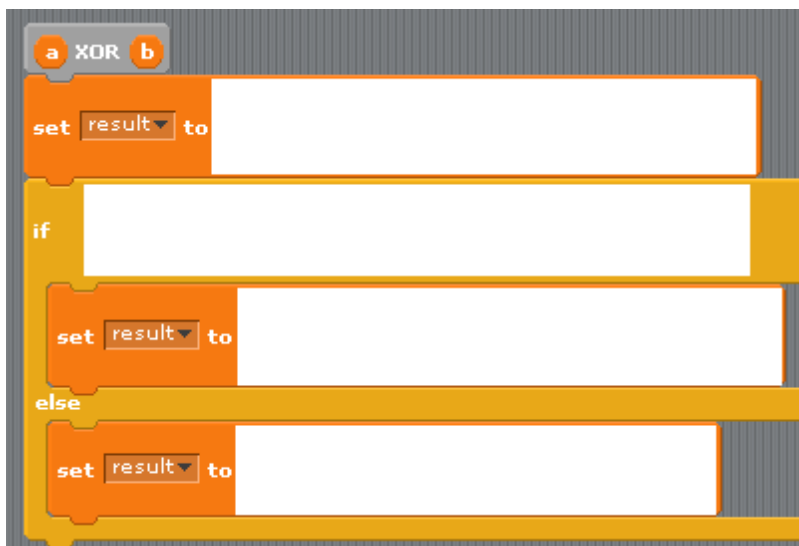
b) using **and/or** but without using **if/else** by filling out ONLY the report blank. *Hint: you may nest blocks together.*
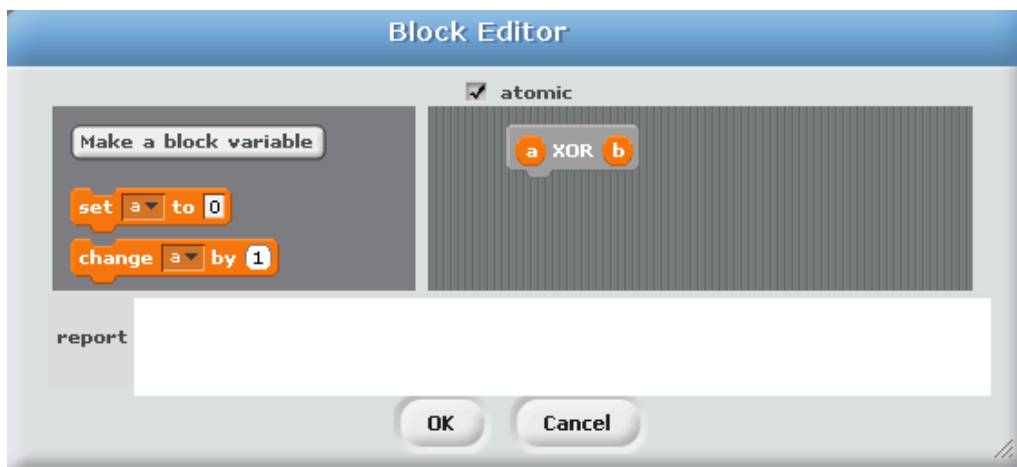
2. *Definition: **Pig Latin** : made-up language formed from English by transferring the initial consonant or consonant cluster of each word to the end of the word and adding "ay" at the end): SCRATCH would become ATCHSCRAY.*
Here is the code for **piglatin** with helpers **vowel?**, **first** and **butfirst** (the latter two you've seen before):
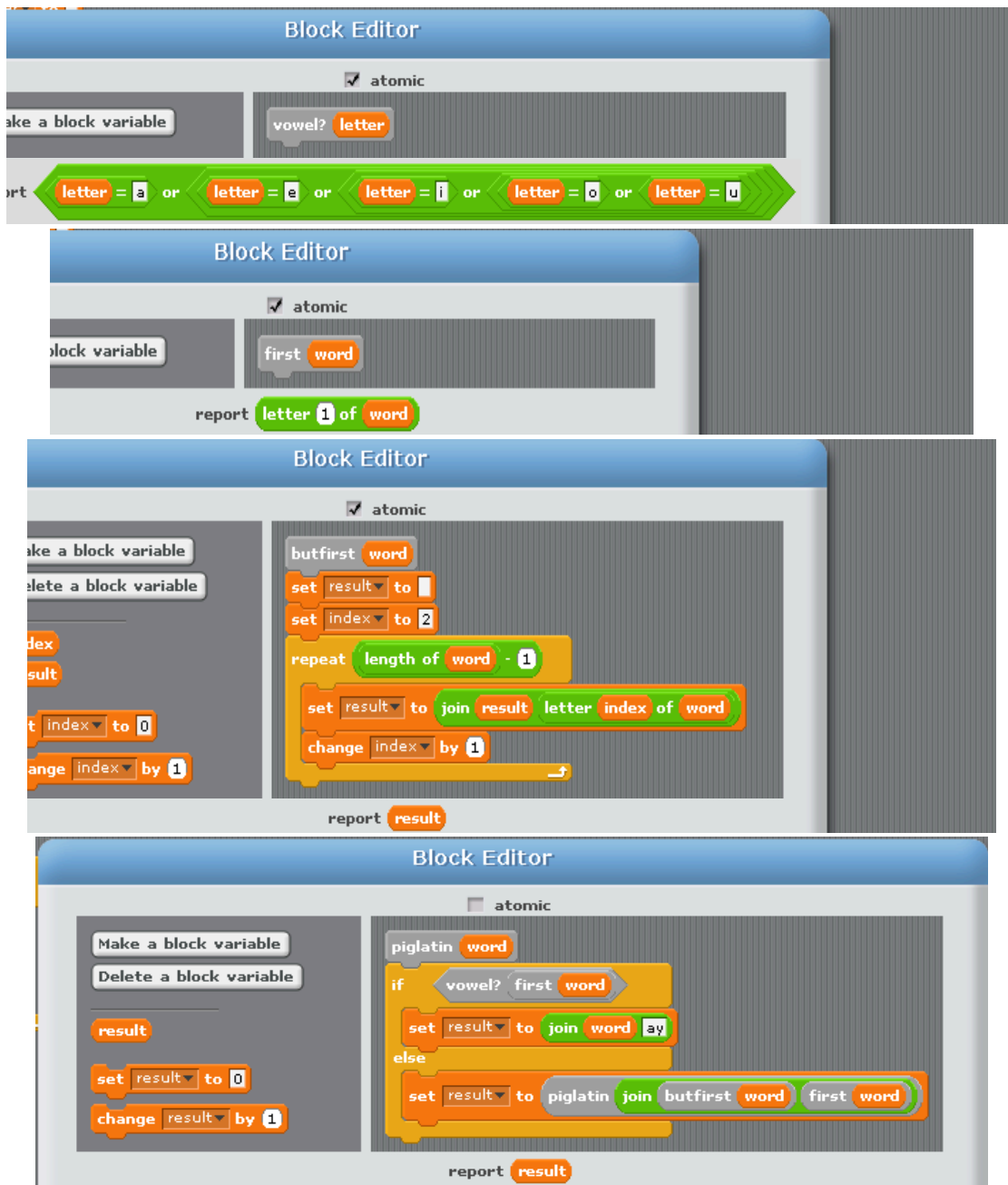
a) What is `piglatin america` ?

b) Will **piglatin** always return successfully? If so, provide a one-sentence argument why it will. If not, what does it do? (circle one: *return incorrect answer*, *crash*, *infinite-loop*), and provide an input word that causes it to do what you circled.

c) Your partner suggests you write **un-piglatin** to take a piglatin word and return the original word that generated it. E.g., `unpiglatin atchscray` should return *scratch*. In one sentence, what's wrong with this idea?
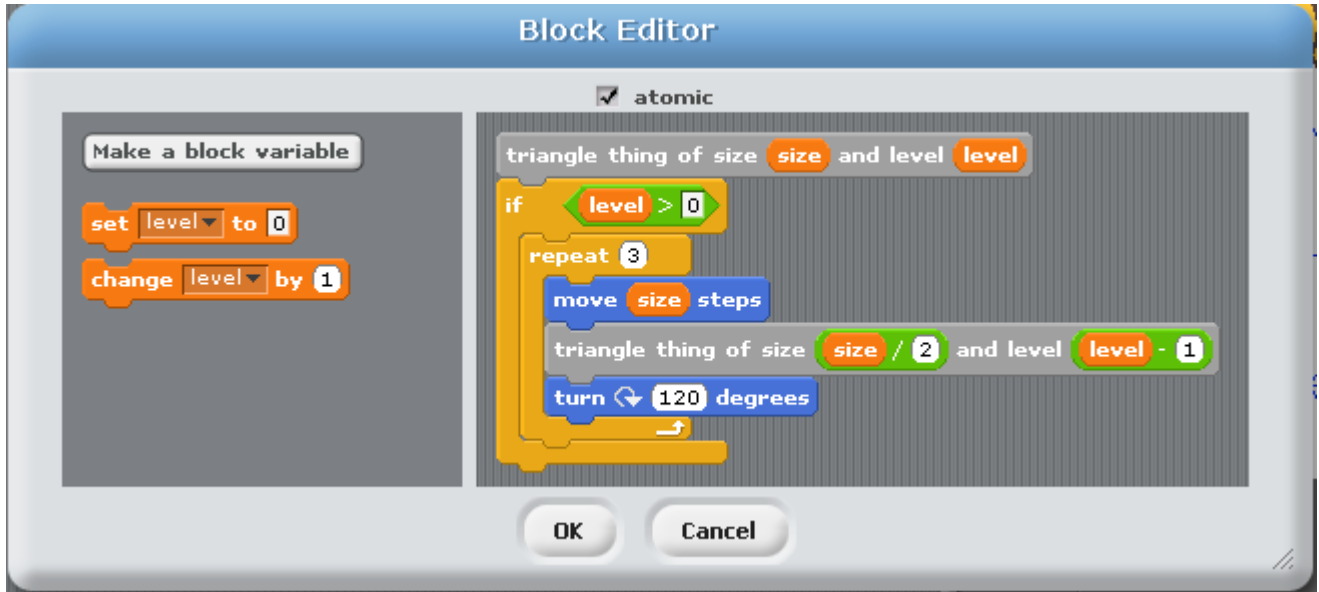
3. Given **fun** defined below:
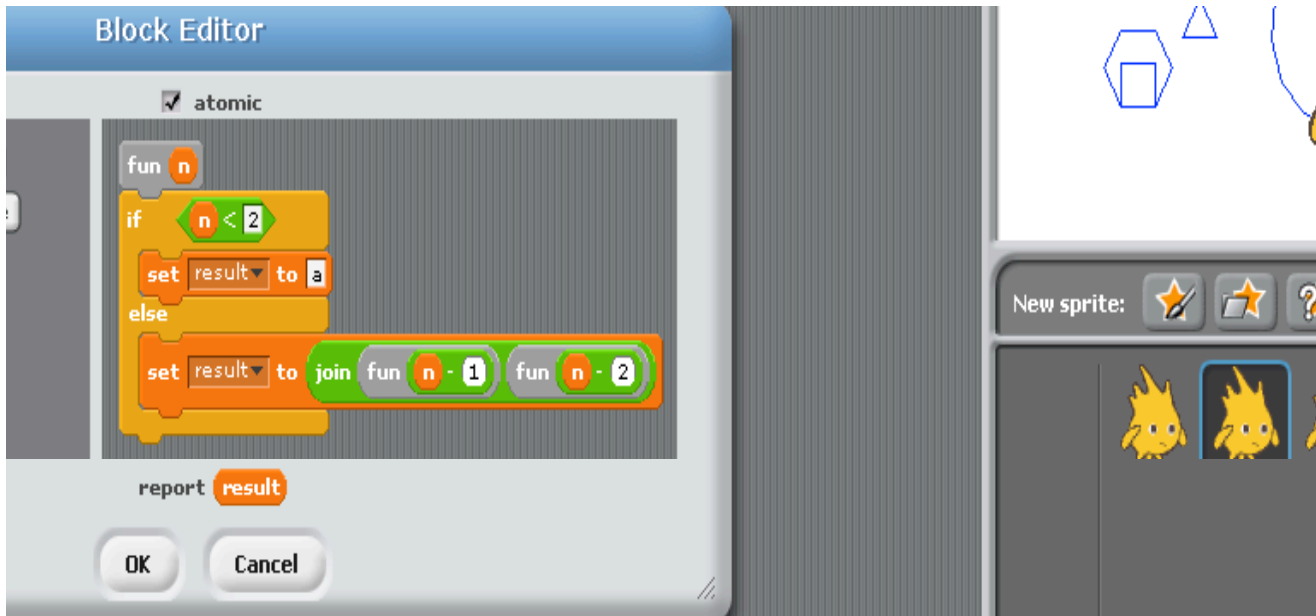
a), how many a's are reported when called with each n value below? Write the number of "a"s in the boxes below.

fun 0	fun 1	fun 2	fun 3	fun 4	fun 5

b) What other function has a similar pattern?

4. Draw the fractals created by this block with size = 240 and level = 1, level = 2 and level = 3.
   The arrow indicates the initial direction and position of your sprite.

Block Editor

☑ atomic

Make a block variable

set level ▼ to 0

change level ▼ by 1

triangle thing of size size and level level

if (level > 0)
  repeat 3
    move size steps
    triangle thing of size (size / 2) and level (level - 1)
    turn ↻ 120 degrees

OK   Cancel

triangle thing of size 240 and level 1	triangle thing of size 240 and level 2
triangle thing of size 240 and level 3	

Name:

CS39n Login: cs39n-____

(you can use the rest of this page as Scratch paper)

Part With Computers:

Instructions:

First, go to the bSpace final assignment, and download the final.sb file. This file has multiple sprite, one sprite for each question in this part of the exam. Do each exercise in the sprite indicated. Please save your scratch file as "FIRSTNAME_LASTNAME.sb". For example, your TA Colleen would save their file as "COLLEEN_LEWIS.sb". Submit this under the bSpace 'final' assignment with the written portion (below).
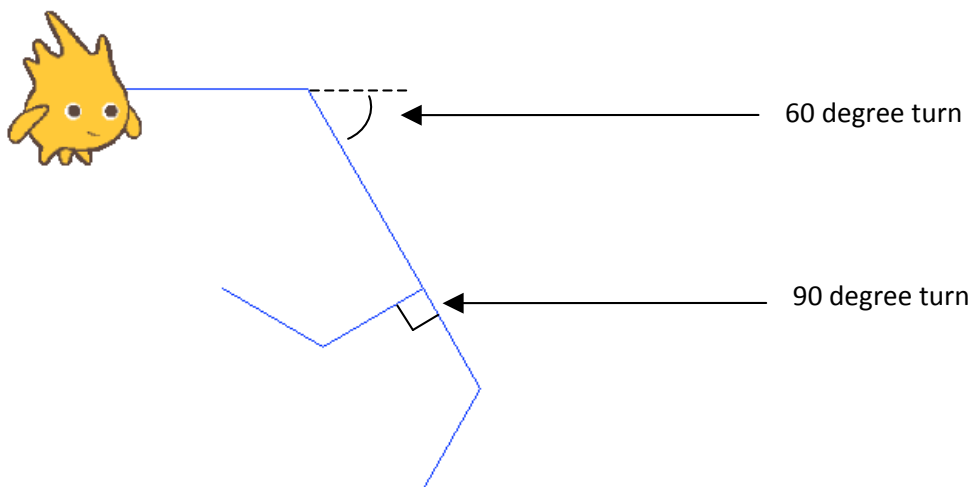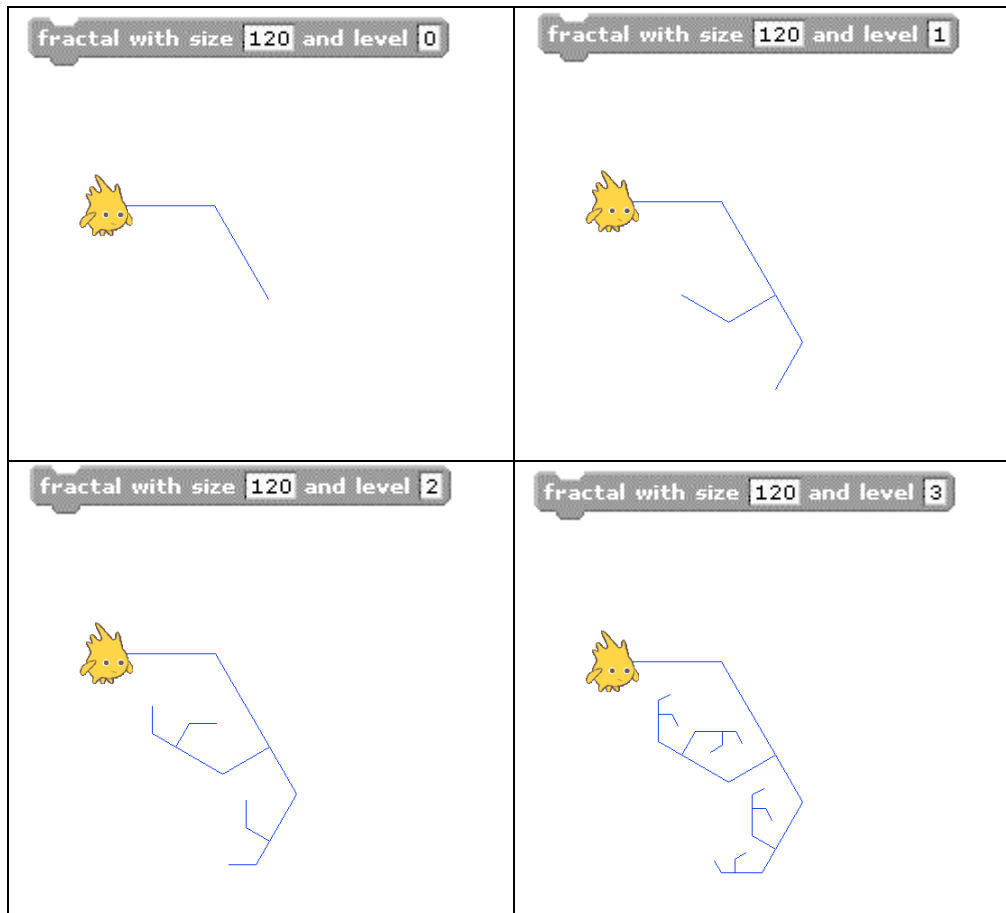
Then, for each sprite and any blocks you modify or create, paste their screenshots together, and print out a physical copy. You will turn these in as well.

5. You're told that the * block will cause the computer to crash if you use it. "No Problem," you say, "I'll just build my own '*x product y*' block using BYOB!" Assume *x* is always a non-negative integer (i.e., 0,1,2,3,...), but *y* can be any real number: (1.283, -3.14, ...). You realize there are two ways to do this, one using recursion and one without.

    a) Write the block '*x nonrecursive-product y*' *without* recursion. Hint: `3*1.7 = 1.7 + 1.7 + 1.7`

    b) Write the block '*x recursive-product y*' *with* recursion. Hint: `3*1.7 = 1.7 + (2 * 1.7)`

6. Write a program to generate the fractal shown below. The images below show the fractal for level=0 level=1, level=2, and level=3. The sprite should turn 60 degrees to generate the initial bend, and the two line segments in the base case are of length 120. Each "bent L" is 1/2 of the length of the one in the level before it. The two drawings of the next level are separated 90 degrees apart.

fractal with size 120 and level 0

fractal with size 120 and level 1

fractal with size 120 and level 2

fractal with size 120 and level 3

60 degree turn

90 degree turn

Instructions: Please type the following into a text editor and submit it under bSpace as well. Turn in a printed copy.

7. *Definition: **democracy** : the practice or principles of social equality.*
   Computers have served as both a democratizing (giving power to the people) **and** anti-democratizing

(concentrating power in the hands of the rich/powerful) influence. Describe the two most significant democratizing uses. Then, describe the two most significant anti-democratizing influences and suggest politically & socially feasible (i.e., things that could actually be implemented in our existing society) ways to counteract it. Here is one example each – remember, you're asked for two examples)

> *Democratizing*: Blogs/cameraphones/youtube have completely transformed information creation; stripping power from mainstream media sources and allowing anyone with something to say an easy, and often free outlet to share opinions, info, etc.

> *Anti-democratizing*: The loss of mom-n-pop stores to massive internet presences like Amazon, who can demand deep discounts from publishers and don't have traditional overhead costs. Solutions that might work: Actively pursue anti-monopoly laws to limit growth. Find a way to work *with* Amazon (a feature they later added called "Marketplace") to incorporate listings from smaller stores.

8. Proposal: Require a subdermal (underneath the skin) implant of a GPS (global positioning system) chip in every human. This GPS chip would have a unique identifier and be able to be queried externally to request its position and identifier number. Therefore, it could help us answer two questions:

   • *Who* is this person? [the chip would return its identifier number]
   • *Where* is a particular person? [the network would return the GPS coordinates of the person in question]

   Write the equivalent of an "environmental impact report" in which you discuss all the aspects of this proposal: benefits, downfalls, possible abuses, etc. Strive for breadth and try to cover as many angles as you can (max 4). To keep this organized, limit every "angle" to one sentence, and preceed it by the aspect category. E.g.,

   > • downfall: implanting it would probably hurt
   > • benefit: people who like pain would probably enjoy the implantation
   > • etc.

9. In the future, we may be able to connect a human brain to a computer. However assume we can only go in one direction. That is, we can either send **output from a computer** to a human brain (computer→human, so the computer could drive your senses or muscles…obviating the need for a display or speakers) or **input data to a computer** from a human brain (human→computer, so you could think something and the computer would know it…obviating the need for a keyboard or mouse). Argue which you'd choose and why, with 4 one-sentence examples of the most amazing applications one could realize with the option you chose.