

CompSci 94 Exam 1 Spring 2019

Given below are the condition possibilities for an if statement

The image shows the 'if' statement menu in Scratch. The menu is titled 'if true is true then'. It has a 'true' button (current value) and an 'else' button. Below these are several condition options:

- nextRandomBoolean
- NOT true
- NOT ???
- BOTH true AND ???
- EITHER true OR ???
- BOTH ??? AND ???
- EITHER ??? OR ???
- Relational (DecimalNumber) { ==, !=, <, <=, >=, > }
- Relational (WholeNumber) { ==, !=, <, <=, >=, > }
- Relational (SThing) { ==, != }
- Relational (MoveDirection) { ==, != }
- Relational (TurnDirection) { ==, != }
- Relational (RollDirection) { ==, != }
- Relational (Key) { ==, != }
- Relational (Color) { ==, != }
- Relational (Paint) { ==, != }
- TextString Comparison

Arrows point from the 'Relational (DecimalNumber)', 'Relational (WholeNumber)', 'Relational (SThing)', 'Relational (MoveDirection)', 'Relational (TurnDirection)', 'Relational (RollDirection)', 'Relational (Key)', 'Relational (Color)', 'Relational (Paint)', and 'TextString Comparison' options to a list of specific conditions:


- ??? < ???
- ??? ≤ ???
- ??? > ???
- ??? ≥ ???
- ??? == ???
- ??? ≠ ???
- ??? == ???
- ??? ≠ ???
- ??? contentEquals ???
- ??? equalsIgnoreCase ???
- ??? startsWith ???
- ??? endsWith ???
- ??? contains ???

Below are the tiles at the bottom of a **procedure**

The image shows the tiles at the bottom of a procedure in Scratch. The tiles are:

- do in order
- count _
- while _
- for each in _
- if _
- do together
- each in _ together
- variable...
- assign
- //comment

Given below are the panda procedures and panda Properties on the bottom right.

 **this.panda**

Procedures **Functions**

group by category ▼

Panda 's Editable Procedures (3)

- ☐ edit **this.panda** standingPose
- ☐ edit **this.panda** sleepingPose
- ☐ edit **this.panda** crawlingPose

Biped 's Editable Procedures (0)

say, think

- this.panda** say text: ???
- this.panda** think text: ???

position

- this.panda** move direction: ???, amount: ???
- this.panda** moveToward target: ???, amount: ???
- this.panda** moveAwayFrom target: ???, amount: ???
- this.panda** moveTo target: ???
- this.panda** place spatialRelation: ???, target: ???

orientation

- this.panda** turn direction: ???, amount: ???
- this.panda** roll direction: ???, amount: ???
- this.panda** turnToFace target: ???
- this.panda** orientTo target: ???
- this.panda** orientToUpright
- this.panda** pointAt target: ???

position & orientation

- this.panda** moveAndOrientTo target: ???

size

- this.panda** setWidth width: ???
- this.panda** setHeight height: ???
- this.panda** setDepth depth: ???
- this.panda** resize factor: ???
- this.panda** resizeWidth factor: ???
- this.panda** resizeHeight factor: ???
- this.panda** resizeDepth factor: ???

appearance

- this.panda** setPaint paint: ???
- this.panda** setOpacity opacity: ???

vehicle

- this.panda** setVehicle vehicle: ???

audio

- this.panda** playAudio audioSource: ???

timing

- this.panda** delay duration: ???

other

- this.panda** straightenOutJoints

 **this.panda**

one shots ▼

this.panda's Properties

Panda panda ← new Panda

Paint = ☐ WHITE

Opacity = 1.0

Vehicle = this

Position = (x: -1.00, y: 1.73, z: -0.10)


Size =

- Width: 0.75
- Height: 1.15
- Depth: 0.53

Reset

Show Joints: ☐

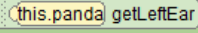
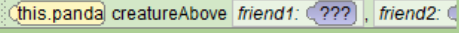
Given below are the panda functions.

 **this.panda**

Procedures **Functions**

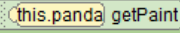
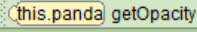
group by category ▼

Panda's Editable Functions (3)

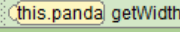
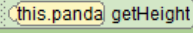
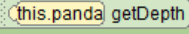
- ☐ edit 
- ☐ edit 
- ☐ edit 

Biped's Editable Functions (0)

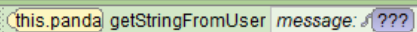
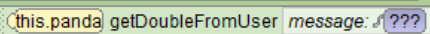
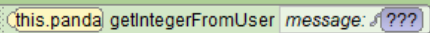
appearance

- 
- 

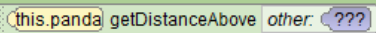
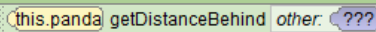
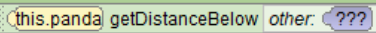
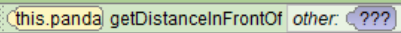
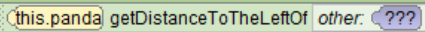
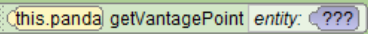
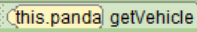
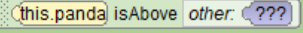
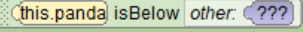
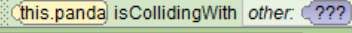
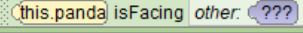
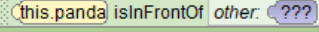
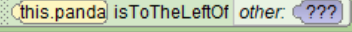
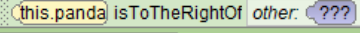
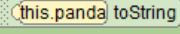
size

- 
- 
- 

prompt user

- 
- 
- 
- 

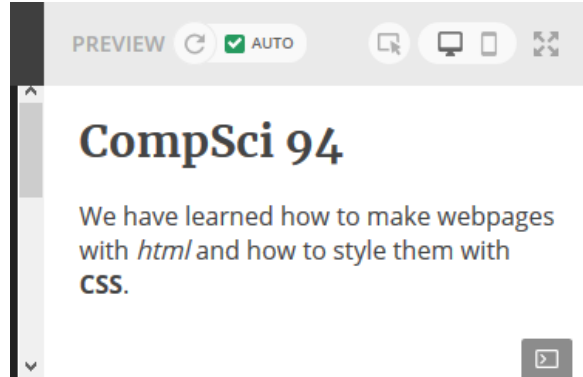
other

- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 

joints

- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
- 

1. (3 pts) Consider the following web page



In the web page, which tags were used on the words *html* and **CSS**?

- a) The `<pre>` tag on the word *html* and the `<i>` tag on the word **CSS**.
- b) The `<i>` tag on the word *html* and the `<pre>` tag on the word **CSS**.
- c) The `` tag on the word *html* and the `` tag on the word **CSS**.
- d) The `` tag on the word *html* and the `` tag on the word **CSS**.

2. (3 pts) Consider the following HTML code that defines some lists.

```
<ol>
<li> ACC</li>
  <ul>
    <li> Duke </li>
  </ul>
<li> SEC </li>
<ol>
<li> Tennessee</li>
</ol>
</ol>
```

Which one of the following is the way this HTML code will be displayed on a web page?

- | | |
|---|--|
| A) <ul style="list-style-type: none">1. ACC<ul style="list-style-type: none">◦ Duke2. SEC<ul style="list-style-type: none">1. Tennessee | B) <ul style="list-style-type: none">1. ACC<ul style="list-style-type: none">◦ Duke2. SEC<ul style="list-style-type: none">◦ Tennessee |
| C) <ul style="list-style-type: none">• ACC<ul style="list-style-type: none">1. Duke• SEC<ul style="list-style-type: none">◦ Tennessee | D) <ul style="list-style-type: none">• ACC<ul style="list-style-type: none">1. Duke• SEC<ul style="list-style-type: none">1. Tennessee |

3. (3 pts) Consider the following html code to display a table.

```
<table>
  <tr><th>tulips</th><th>roses</th>
  <td>pansies</td><td>daffodils</td></tr>
</table>
```

Which one of the following is the way this HTML code will be displayed on a web page?

- A)

tulips	roses
pansies	daffodils
- B)

tulips	roses	pansies	daffodils
--------	-------	---------	-----------
- C)

tulips	roses
pansies	daffodils
- D)

tulips	roses	pansies	daffodils
--------	-------	---------	-----------

4. (3 pts) Consider the following HTML code:

```
<h2 id="change"> Turtles</h2>
```

```
<h2> Seagulls </h2>
```

and .css code:

```
#change
{ color: blue; }
#thing
{ color: yellow; }
```

The default color with NO CSS is black. How are the words Turtles and Seagulls displayed?

- A) Turtles and Seagulls are both in black
- B) Turtles is in blue and Seagulls is in black
- C) Turtles is in blue and Seagulls is in yellow
- D) Turtles and Seagulls are both in blue.

5. (3 pts) Consider the following HTML code: and .css code:

```
<h2 class="something"> Turtles</h2>
```

```
h2 { color: green; }
```

```
<h2> Seagulls </h2>
```

```
.something {color: purple; }
```

The default color with NO CSS is black. How are the words Turtles and Seagulls displayed?

- A) Turtles and Seagulls are both in black
 - B) Turtles is in purple and Seagulls is in green
 - C) Turtles is in green and Seagulls is in purple
 - D) Turtles is in purple and Seagulls is in black.
6. (4 pts) A six digit hexadecimal number represents a color. A color can also be represented as a three-tuple of numbers representing the red, green and blue components (or RGB) of the number. SHOW YOUR WORK.
- A. Convert the hexadecimal color number #2E0A3C into RGB: (, ,)

 - B. Convert the color in RGB (35, 12, 27) to a six-digit hexadecimal color number:

7. (3 pts) Consider the pig shown below. Which line of code has the pig change position from the top picture to the bottom picture?



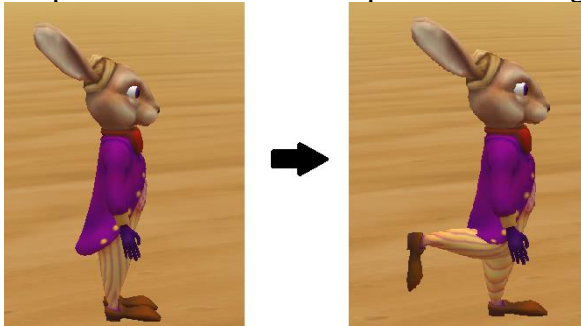
- A) `do in order`
`this.pig roll RIGHT, 0.125 add detail`
 B) `this.pig roll RIGHT, 0.25 add detail`
 C) `this.pig roll LEFT, 0.125 add detail`
 D) `this.pig roll LEFT, 0.25 add detail`

8. (3 pts) Consider the pig shown below. Which do in order block of code has the pig change position from the top picture to the bottom picture?



- A) `do in order`
`this.pig turn RIGHT, 0.25 add detail`
`this.pig move BACKWARD, 1.0 add detail`
 B) `do in order`
`this.pig move BACKWARD, 1.0 add detail`
`this.pig turn RIGHT, 0.25 add detail`
 C) `do in order`
`this.pig move FORWARD, 1.0 add detail`
`this.pig turn RIGHT, 0.25 add detail`
 D) `do in order`
`this.pig turn RIGHT, 0.25 add detail`
`this.pig move FORWARD, 1.0 add detail`

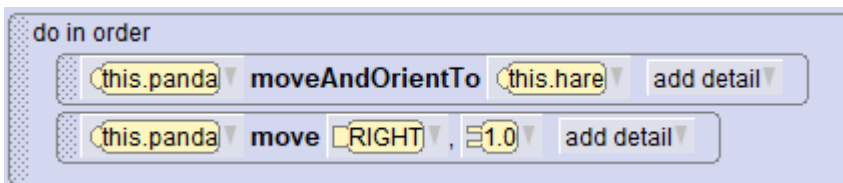
9. (3 pts) Consider the following pictures in which the marchHare changes from the picture on the left to the picture on the right.



Which one of the following lines of code is executed to make this change in the marchHare?

- A) `this.marchHare.getRightHip.turn(FORWARD, 0.125)` add detail
- B) `this.marchHare.getRightHip.turn(BACKWARD, 0.125)` add detail
- C) `this.marchHare.getRightHip.turn(FORWARD, 0.25)` add detail
- D) `this.marchHare.getRightHip.turn(BACKWARD, 0.25)` add detail

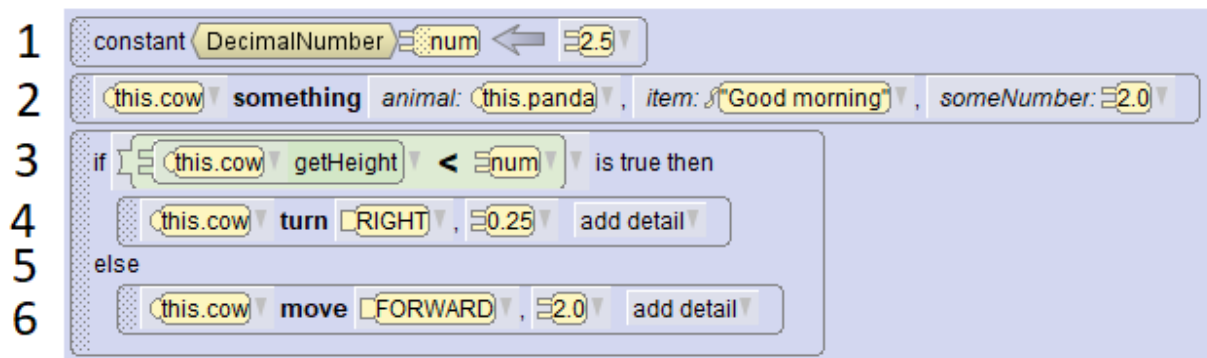
10. (3 pts) Consider a program with a hare and a panda as shown in the figure on the left. Which one of the following will be the position of the panda and hare after this code executes?



Start

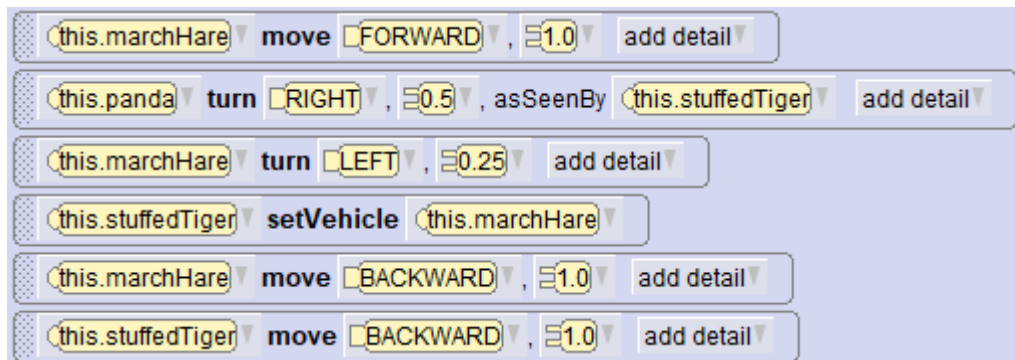
- A)
- B)
- C)
- D)

11. (13 pts) Consider the following Alice code in which the lines are numbered.



- A) In line 2, what is the name of the procedure?
- B) In line 2, what is the type of the procedure?
- C) In line 2, list the word(s) that are arguments.
- D) In line 2, list the word(s) that are parameters.
- E) In line 2, I can put the words bunny or pig in place of the word panda, but I cannot put camel or dalmatian. What are the types of all the parameters in this line?
- F) What must be true in order for line 6 to execute?

12. (4 pts) Consider the following program that has the three objects: `stuffedTiger`, `marchHare` and `panda` (shown below from left to right), and given code. The world has been setup as shown below. The `stuffedTiger` is **exactly 1.0 meter** from the `marchHare`, and the `panda` is **exactly 1.0 meter** from the `marchHare`.



The diagram below is looking from above over the scene. The `stuffedTiger` is represented by the S, the `marchHare` is represented by the M, and the `panda` is represented by the P. The animals are facing the bottom of the page. Using the diagram below, draw the path of `stuffedTiger` and `marchHare` as a solid line and the path of `panda` as a dashed line.

S M P

13. (4 pts) Consider the following **Panda** Mystery procedure.

declare procedure **mystery** with parameters: DecimalNumber \Rightarrow amount, DecimalNumber \Rightarrow value, Paint \Rightarrow someColor

do in order

- if BOTH $\{ \text{amount} > \text{value} \}$ AND $\{ \text{someColor} \neq \text{RED} \}$ is true then
 - if $\{ \text{someColor} == \text{BLUE} \}$ is true then
 - this say "frog" add detail
 - else
 - this say "horse" add detail
- else
 - if EITHER $\{ \text{amount} < 5.0 \}$ OR $\{ \text{someColor} == \text{GREEN} \}$ is true then
 - this say "cat" add detail
 - else
 - this say "dog" add detail

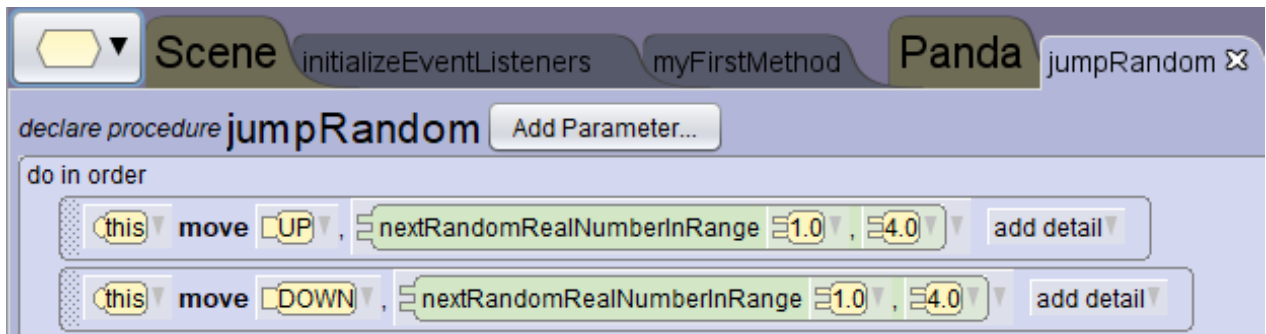
A) What does the **panda say** when the following line of code is run?

this.panda **mystery** amount: 1.6, value: 2.8, someColor: BLUE

B) What does the **panda say** when the following line of code is run?

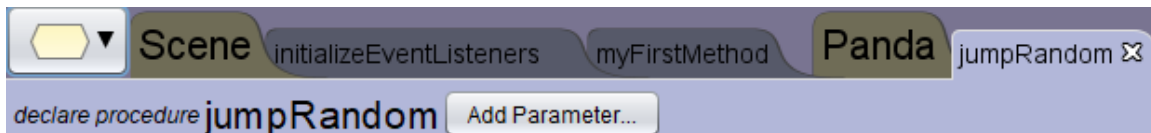
this.panda **mystery** amount: 5.0, value: 3.2, someColor: GREEN

14. (4 pts) Consider the following Panda JumpRandom procedure for the panda to jump a random amount each time the procedure is called.

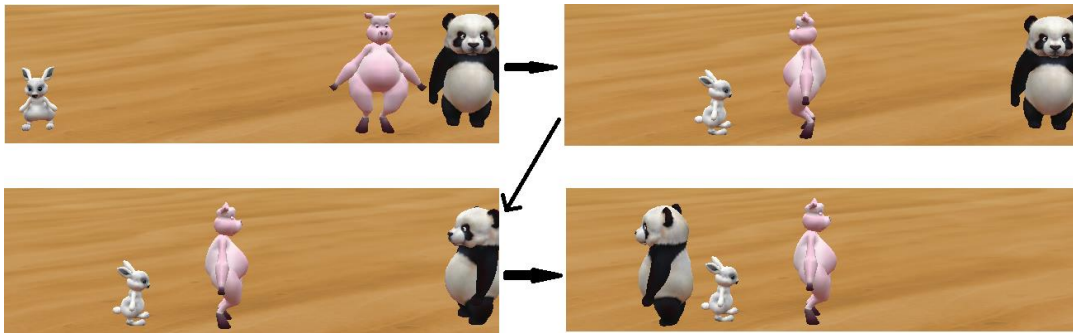


When you run the program and call the jumpRandom procedure with the panda standing on the ground, the panda moves up and then moves down halfway through the ground. The panda is supposed to land on the ground exactly where it was when it started the jump.

- A) Explain why this code does not work correctly.
- B) Give the correct code below for the panda jumpRandom procedure below. Do not add any parameters.



15. (14 pts) Consider the following Alice project that has three objects: bunny, pig and panda.



The program starts as shown in the top left figure above with the bunny, pig and panda standing in a line all facing front. Write code to do the following in this order.

When you move the animals you do not need to move their legs, just move them.

- The bunny and pig turn to face each other at the same time, then the pig moves about half the distance between the pig and bunny, then the bunny moves stopping 0.5 units in front of the pig (see top right picture).
- The panda turns to face the pig and the panda says hello. Then the pig turns around to face the panda. (bottom left picture).
- The panda jumps over the pig and bunny (not touching either one), landing past the bunny. (bottom picture right).
- Both the bunny and pig disappear. Then the panda turns to face forward and then turns completely red.

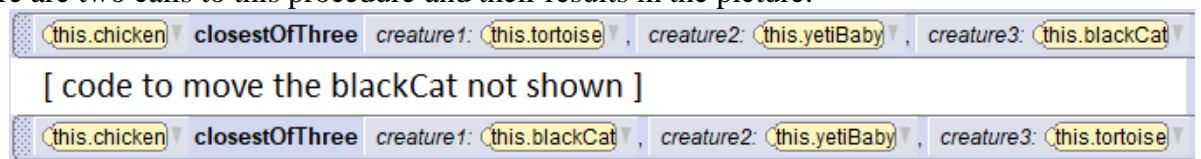
```
declare procedure myFirstMethod
do in order
```

(extra page)

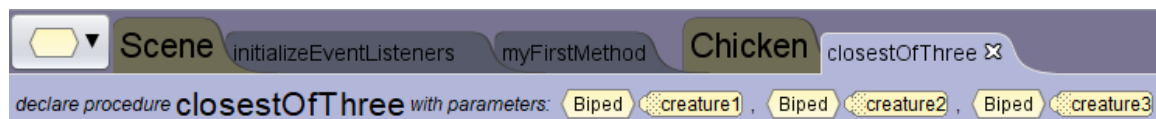
16. (10 pts) Complete the following Chicken **procedure** called `closestOfThree` whose header is shown below. This procedure has three **Biped** parameters, named **creature1**, **creature2**, and **creature3**. This procedure has the object that is closest to the chicken (of the three), say “I’m closest”. (note the blackCat is a Biped)



Here are two calls to this procedure and their results in the picture.



Complete the code below.



(extra page 1, must turn in)

(extra page 2, must turn in)