CODING – PART 4

BIG IDEAS:

- An ***if* statement** is used to execute code only if a specified condition is met
- An ***if…else* statement** is used to execute one of two blocks of code, depending on whether or not a specified condition is met

LEARNING GOALS AND SKILL DEVELOPMENT:
You know you have met the goals for this lesson when you can:

EMERGING

LEARNING GOALS	ANCHOR QUESTIONS
Predict the output of a program involving an *if…else* statement	1, 2

SKILL BUILDING QUESTIONS			
1	2		

EVOLVING

LEARNING GOALS	ANCHOR QUESTIONS
Explain the function of a program involving an *if…else* statement and predict its output	3, 4
Write a program involving an *if…else* statement	5
Write a program involving an *if…else* statement and a loop	6

SKILL BUILDING QUESTIONS			
3	4	5	6

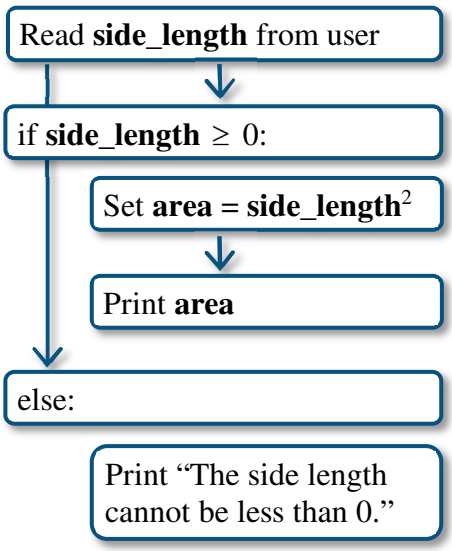
EXTENDING

LEARNING GOALS	ANCHOR QUESTIONS
Predict the output of a program involving an *if…else if…else* statement	7
Write a program involving an *if…else if…else* statement	8

SKILL BUILDING QUESTIONS			
7	8		

BUILD YOUR SKILLS

1. The blocks of code shown on the right are used to calculate the area of a square.



a) What will be the output of the program if it is run with a user input of 5?

b) What will be the output of the program if it is run with a user input of 0?

c) What will be the output of the program if it is run with a user input of −1?

2. Consider the Python program on the right.

```
1  # Read guess value from user.
2  guess=int(input("Guess my age: "))
3
4  # Evaluate guess and print message.
5  if guess==15:
6      print("You got it!")
7  else:
8      print("Sorry, that is incorrect!")
```

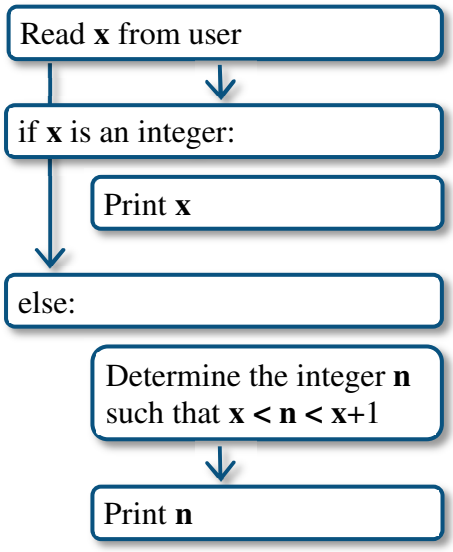
a) Why is a double equal sign used in line 5?

b) State the output of the program if the user enters 14.

c) What will happen if the user enters a value of 15.5? Explain.

d) Would the program still run if lines #7 and #8 were omitted? Explain.

3. Consider the program shown on the right.



a) Describe what the program does.

b) State the output that results when each of the following values is assigned to x.

i) 8 ii) −3 iii) 4.7 iv) −6.2 v) π

4. Consider the Python program shown below.

```
1  import math
2
3  x=float(input("Please enter a value: "))
4
5  if x>=0:
6      rootx=math.sqrt(x)
7      print("The square root of",x,"is",rootx)
8  else:
9      print("The square root of",x,"is not a real number.")
```

a) What is the purpose of the *if…else* statement used in this program?

b) Describe what the program does.

c) What will be the output of the program if it is run with a user input of -4.0?

5. A Python program is to be written to calculate unknown side lengths in right triangles.

a) Create a program that calculates the length of the hypotenuse based on the lengths of the other two sides entered by the user.

b) Create a program that calculates the length of the unknown leg based on the lengths of the hypotenuse and the other leg entered by the user.

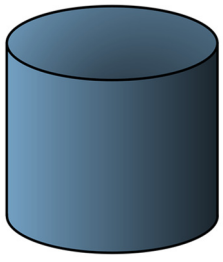
c) Using your programs from parts (a) and (b), create a new program that allows the user to first choose whether the hypotenuse or another side length is to be calculated.

d) Modify your program from part (c) to display the message, "`The hypotenuse must be longer than each leg`," when applicable.

6. The surface area of a cylinder with radius r and height h can be found using the formula $A = 2\pi r^2 + 2\pi rh$.

a) Using a *for loop*, create a Python program to display the surface area of a cylinder with radius 1 cm for height values from 0 cm through 10 cm (integers only).

b) Using a *for loop*, create a Python program to display the surface area of a cylinder with height 1 cm for radius values from 0 cm through 10 cm (integers only).



c) Modify your program in part (a) to allow the user to enter the fixed radius value and modify your program in part (b) to allow the user to input the fixed height value.

d) Using an *if…else* statement and your programs from part (c), create a new program that allows the user to first choose whether the radius or height is fixed.

e) Does the relationship between height and surface area appear to be linear? How about the relationship between radius and surface area? Explain.

7. The *elif* statement in Python stands for "else if" and allows multiple conditions to be checked in sequence. An *elif* statement is executed only if the preceding *if* or *elif* condition is false. Consider the example shown below.

```
1  # Read luggage mass from user.
2  mass=float(input("Please enter the mass of your luggage in pounds: "))
3
4  # Print message.
5  if mass>120:
6      print("Sorry, your luggage is too heavy.")
7  elif mass>50:
8      print("Your luggage is within limits, but a $20 charge applies.")
9  else:
10     print("Your luggage is within limits and no additional charge applies.")
```

Predict the output of the program for each of the following masses.

a) 28 lb b) 75 lb c) 125 lb d) 120 lb

8. The height of an object, in metres, t seconds after it is launched straight up into the air is given by the equation $h = -4.9t^2 + bt + c$, where b represents the ball's initial speed in metres per second and c represents the ball's initial height in metres. Using the *elif* statement described in question #7, write a Python program that displays the height of the ball every second from the instant it is thrown until it hits the ground. Design your program such that the following criteria are satisfied:

- The user enters the initial speed and the initial height.
- The format of the output is as follows: `After 2 seconds, the height is 30.87 m`.
- Instead of displaying negative height values, the output states that the ball is on the ground.
- The output does not contain grammatical errors, such as "`After 1 seconds,….`"

CHECK YOUR UNDERSTANDING

1. a) `25` b) `0` c) `The side length cannot be less than 0.`

2. a) A double equal sign must be used when checking if two values are equal. A single equal sign is used for assigning values.

b) `Sorry, that is incorrect!`

c) The program will return an error since an integer is expected and not a floating point number.

d) Yes. *If* statements do not require an *else* component. If the user enters a value of 15, the output will be "`You got it!`" Otherwise, no message will be displayed for the output.

3. a) The user is prompted to enter a value. If an integer is entered, that integer will be displayed as the program's output. Otherwise, the program will determine the integer between the entered value and 1 more than that value, which it will display as the output. The program essentially rounds non-integer values up to the closest integer.

b) i) 8 ii) –3 iii) 5 iv) –6 v) 4

4. a) The *if…else* statement is used to differentiate between user inputs that are greater than or equal to zero and those that are negative.

b) The user is prompted to enter a value. If the user enters a value greater than or equal to zero, that value's square root is computed and displayed. If a negative value is entered, a message stating that the value's square root is not a real number is displayed.

c) `The square root of -4.0 is not a real number.`

5. a)

```
# Import math library (for square root).
import math

# Read values for leg lengths.
leg1=float(input("Enter the length of leg 1: "))
leg2=float(input("Enter the length of leg 2: "))

# Calculate the length of the hypotenuse.
hyp=math.sqrt(leg1**2+leg2**2)

# Print hypotenuse length.
print("The length of the hypotenuse is",hyp)
```

ANSWERS

ANSWERS

b)

```
# Import math library (for square root).
import math

# Read values for hypotenuse and leg lengths.
hyp=float(input("Enter the length of the hypotenuse: "))
leg1=float(input("Enter the known leg length: "))

# Calculate the unknown leg length.
leg2=math.sqrt(hyp**2-leg1**2)

# Print the calculated leg length.
print("The length of the other leg is",leg2)
```

c)

```
# Import math library (for square root).
import math

# Select type of problem.
type=input("Are you finding the hypotenuse? Enter y or n: ")

# Find unknown side length.
if type=="y":
    # Read values for leg lengths.
    leg1=float(input("Enter the length of leg 1: "))
    leg2=float(input("Enter the length of leg 2: "))

    # Calculate the length of the hypotenuse.
    hyp=math.sqrt(leg1**2+leg2**2)

    # Print hypotenuse length.
    print("The length of the hypotenuse is",hyp)

else:
    # Read values for hypotenuse and leg lengths.
    hyp=float(input("Enter the length of the hypotenuse: "))
    leg1=float(input("Enter the known leg length: "))

    # Calculate the unknown leg length.
    leg2=math.sqrt(hyp**2-leg1**2)

    # Print the calculated leg length.
    print("The length of the other leg is",leg2)
```

ANSWERS

d)

```
# Import math library (for square root).
import math

# Select type of problem.
type=input("Are you finding the hypotenuse? Enter y or n: ")

# Find unknown side length.
if type=="y":
    # Read values for leg lengths.
    leg1=float(input("Enter the length of leg 1: "))
    leg2=float(input("Enter the length of leg 2: "))

    # Calculate the length of the hypotenuse.
    hyp=math.sqrt(leg1**2+leg2**2)

    # Print hypotenuse length.
    print("The length of the hypotenuse is",hyp)

else:
    # Read values for hypotenuse and leg lengths.
    hyp=float(input("Enter the length of the hypotenuse: "))
    leg1=float(input("Enter the known leg length: "))

    # Compare side lengths.
    if hyp<leg1:
        print("The hypotenuse must be longer than each leg.")

    else:
        # Calculate the unknown leg length.
        leg2=math.sqrt(hyp**2-leg1**2)

        # Print the calculated leg length.
        print("The length of the other leg is",leg2)
```

6. a)

```
# Import math library (for pi).
import math

# Set fixed radius value.
r=1

# Calculate and print surface areas.
for h in range(11):
    A=2*math.pi*r**2+2*math.pi*r*h
    print(h,"---",A)
```

b)

```
# Import math library (for pi).
import math

# Set fixed height value.
h=1

# Calculate and print surface areas.
for r in range(11):
    A=2*math.pi*r**2+2*math.pi*r*h
    print(r,"---",A)
```

c)

```
# Import math library (for pi).
import math

# Set fixed radius value.
r=float(input("Enter the radius: "))

# Calculate and print surface areas.
for h in range(11):
    A=2*math.pi*r**2+2*math.pi*r*h
    print(h,"---",A)
```

```
# Import math library (for pi).
import math

# Set fixed height value.
h=float(input("Enter the height: "))

# Calculate and print surface areas.
for r in range(11):
    A=2*math.pi*r**2+2*math.pi*r*h
    print(r,"---",A)
```

ANSWERS

d)

```
# Import math library (for pi).
import math

# Select type of problem.
type=input("Would you like to use a fixed radius or height? Enter r or h: ")

# Calculate and display surface areas.
if type=="r":
    # Set fixed radius value.
    r=float(input("Enter the radius: "))

    # Calculate and print surface areas.
    for h in range(11):
        A=2*math.pi*r**2+2*math.pi*r*h
        print(h,"---",A)

else:
    # Set fixed height value.
    h=float(input("Enter the height: "))

    # Calculate and print surface areas.
    for r in range(11):
        A=2*math.pi*r**2+2*math.pi*r*h
        print(r,"---",A)
```

e) The relationship between height and surface area is linear since the surface area increases at a constant rate as the height increases. The relationship between radius and surface area is not linear since the surface area does not increases at a constant rate as the radius increases (due to the r^2 term).

7. a) `Your luggage is within limits, and no additional charge applies.`

b) `Your luggage is within limits, but a $20 charge applies.`

c) `Sorry, your luggage is too heavy.`

d) `Your luggage is within limits, but a $20 charge applies.`

8.

```
# Set initial values.
t=0
h=0

# Read initial speed and height values.
b=float(input("Enter the initial speed in metres per second: "))
c=float(input("Enter the initial height in metres: "))

# Calculate and display heights.
while h>=0:
    h=round(-4.9*t**2+b*t+c,2)

    if t==1:
        print("After 1 second, the height is",h,"m.")

    elif h>0:
        print("After",t,"seconds, the height is",h,"m.")

    else:
        print("After",t,"seconds, the ball is on the ground.")

    t=t+1
```