CS 171 - Lab 4

Professor Mark W. Boady and Professor Adelaida Medlock

Content by Professor Mark Boady

Detailed instructions to the lab assignment are found in the following pages.

- Complete all the exercises and type your answers in the space provided.

What to submit:

- Lab sheet in PDF

Submission must be done via Gradescope

- Please make sure you matched all questions in Gradescope
- We only accept submissions via Gradescope (not via email or Slack).

Students' Names: Tony Kabilan Okeke

User IDs (abc123): tko35

Possible Points: 86

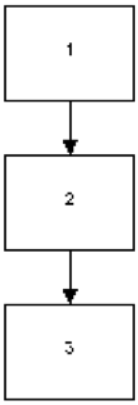
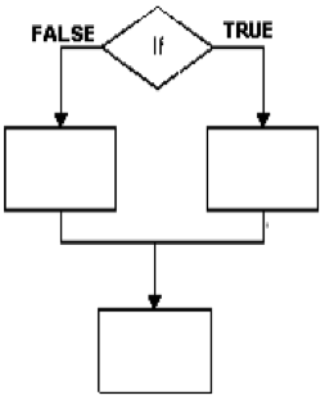
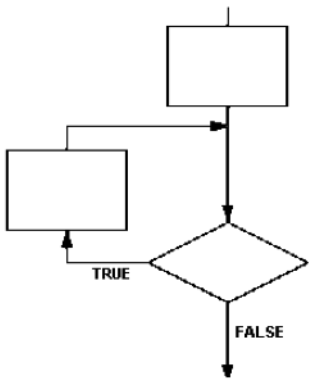
Your score out of 86:

Lab Grade on 100% scale:

Graded By (TA Signature):

Question 1: 2 points

Review the following three Flow Charts. Each describes a type of **Programming Structure**.

Sequence Structure	Branching Structure	Looping Structure
		

(a) (1 point) Which structure best describes the types of Python programs you have written so far?
Sequence Structure

(b) (1 point) Which structure allows the programmer to create code that decides what code is executed?
Branching Structure

FYI: Conditional operators, also known as relational operators, are used to compare the relationship between two operands. Expressions whose result can only be **true** or **false** are known as **Boolean expressions**.

Question 2: 6 points

State the meaning of each of the following conditional operators. If you are not sure of the meaning of any symbol, create some example expressions, type them into the Python interpreter and examine the results.

(a) (1 point) **<** `less than`

(b) (1 point) **>** `greater than`

(c) (1 point) **<=** less than or equal to

(d) (1 point) **>=** greater than or equal to

(e) (1 point) **!=** not equal to

(f) (1 point) **==** equal to

Question 3: 9 points

What is the result of each of the following expressions?

Assume:

```
x = 4
y = 5
z = 4
```

(a) (1 point) `x > y` False

(b) (1 point) `x < y` True

(c) (1 point) `x == y` False

(d) (1 point) `x != y` True

(e) (1 point) `x >= z` True

(f) (1 point) `x <= z` True

(g) (1 point) `x + y > 2 * x` True

(h) (1 point) `y * x - z != 4 % 4 + 16` False

(i) (1 point) `pow(x, 2) == abs(-16)` True

Question 4: 4 points

What is the result of the following expressions?

Assume:

```
word1 = "hello"
word2 = "good-bye"
```

(a) (1 point) `word1 == word2` False

(b) (1 point) `word1 != word2` True

(c) (1 point) `word1 < word2` False

(d) (1 point) `word1 >= word2` True

Question 5: 2 points

How do the conditional operators work when the operands are strings?
Strings are compared character by character in Lexicographic order

Question 6: 2 points

What are the two possible results the expressions in questions 4?
The statements in question 4 can either be True or False

FYI: We can use **logical operators** to determine logic between conditions (relational expressions).

Question 7: 3 points

Sometimes you want to test more than one condition to determine which code segment should be executed. You can use the following **logical operators** to create **compound conditions**. Examine each operator and a sample of its use. Provide an explanation of how each operator works.

(a) (1 point) `(age >= 17) and (hasLicense = = true)`
The `and` operator returns a `True` value if both of its operands are true, and a False otherwise. In the sample above, the statement is true Only when `age >= 17` and `hasLicense == true` are both true.

(b) (1 point) `(cost < 20.00) or (shipping = = 0.00)`
The `or` operator returns a True value when either (or both) of its operands are true. In the sample above, the statement is true when either `cost < 20.00` or `shipping == 0.00` are true.

(c) (1 point) `not (credits> 120)`
The `not` operator negates the value of its operand – Trues become False and vice versa. In the sample above, the statement would be True if `credits < 120` is False and vice versa.

Question 8: 3 points

Assume the value of the variable **numBooks** is **40**. State the values of each of the Boolean expression.

(a) (1 point) `(numBooks > 5) and (numBooks < 100)` True

(b) (1 point) `(numBooks < 5) or (numBooks > 100)` False

(c) (1 point) `not(numBooks * 10 == 100)` True

Question 9: 2 points

Assign a value to **num1** and **num2**. Write a Boolean expression that tests if the value stored in the variable **num1** is equal to the value stored in the variable **num2**

```
In [1]: num1 = 10

In [2]: num2 = 42

In [3]: num1 == num2
Out[3]: False
```

Question 10: 6 points

A **Truth Table** shows the result of a Boolean operator based on its inputs.

Below is the Truth Table for **`and`**

A	B	A and B
True	True	True
True	False	False
False	True	False
False	False	False

(a) (4 points) Draw the Truth Table for **`or`**

A	B	A or B
True	True	True
True	False	True
False	True	True
False	False	False

(b) (2 points) Draw the Truth Table for **`not`** *Hint:* This table is smaller.

A	not A
True	False
False	True

Question 11: 2 points

Assign a value to the variables listed in this problem (**`time`**, **`maxTime`**, **`cost`**, and **`maxCost`**). Write a Boolean expression that tests if the value stored in the variable **`time`** is less than the value stored in the variable **`maxTime`** or if the value stored in the variable **`cost`** is less than the value stored in the variable **`maxCost`**.

```
In [7]: time = 30

In [8]: maxTime = 100

In [9]: cost = 269

In [10]: maxCost = 300

In [11]: (time < maxTime) or (cost < maxCost)
Out[11]: True
```

Question 12: 2 points

Assign a value to **`weight`** and **`cost`**. Write a Boolean expression that tests if the value stored in **`weight`** is less than 10 and the value store in **`cost`** is not greater than 20.00

```
In [12]: weight = 300

In [13]: cost = 20

In [14]: (weight < 10) and not (cost > 20.00)
Out[14]: False
```

Question 13: 2 points

Review the following code block.

```
grade = 95
if grade >= 94:
    print("Excellent!")
```

(a) (1 point) What is the output of the program listed above?
The program prints 'Excellent!' in the console

(b) (1 point) What would the program print if the value stored in the variable of **grade** was 90?
The program would not print anything.

Question 14: 5 points

Enter and execute the following code. Use various values for the cost and the sale price.

```
1  originalPrice = float (input ("Enter the original cost of the time: "))
2  salePrice = float (input ("Enter the sale price: "))
3  percentOff = int (( originalPrice - salePrice )/ originalPrice * 100)
4  print ("Original Price: $" ,format( originalPrice , ".2f") , sep ="")
5  print ("Sale Price: $" ,format( salePrice , ".2f") , sep ="")
6  print ("Percent Off: " ,format( percentOff , "d") ,"%" , sep ="")
7  if ( percentOff >= 50):
8     print ("You got a great sale!")
```

Explain what line of the above program does.

(a) (1 point) Line 1
Collects user input, converts the string provided to a float and stores the value in the variable `originalPrice`

(b) (1 point) Line 3
Compute the percent discount on the item using the values of `originalPrice` and `salePrice`. Then convert the value (float) to an integer – returns only integer portion – and store the integer value in `percentOff`

(c) (1 point) Line 4
Print the string "Original Price: $" followed by the value of `originalPrice` rounded to two decimal places (`.2f`) with no spaces in-between.

(d) (1 point) Line 6
Print the string "Percent Off: " followed by the value of `percentOff` (an integer) followed by the percent symbol (%) with no spaces in-between.

(e) (1 point) Line 7 – 8
Line 7 checks if the value of `percentOff` is greater than or equal to 50. If it is (the expression `percentOff >= 50` is true), the program prints "You got a great sale!" in the console.

Question 15: 2 points

Revise the program in Question 14. If the percent off is 50% or more print "Congratulations!" in addition to what is already printed. Use a second print statement to do this. Do you have to tab in the second print statement? Yes, the second print statement must be at the same indentation depth as the first one.

```
originalPrice = float (input ("Enter the original cost of the time: "))
salePrice = float (input ("Enter the sale price: "))
percentOff = int (( originalPrice - salePrice )/ originalPrice * 100)
print ("Original Price: $" ,format( originalPrice , ".2f") , sep ="")
print ("Sale Price: $" ,format( salePrice , ".2f") , sep ="")
print ("Percent Off: " ,format( percentOff , "d") ,"%" , sep ="")
if ( percentOff >= 50):
   print ("You got a great sale!")
   print ("Congratulations!")
```

Question 16: 2 points

Further revise the program in Question 14 so that it prints "Done!" when the program is complete – no matter what the percent off is. How does the placement of this line of code differ from the placement of the code created for Question 15? This line does not need to be indented since it is not part of the if statement.

```
originalPrice = float (input ("Enter the original cost of the time: "))
salePrice = float (input ("Enter the sale price: "))
percentOff = int (( originalPrice - salePrice )/ originalPrice * 100)
print ("Original Price: $" ,format( originalPrice , ".2f") , sep ="")
print ("Sale Price: $" ,format( salePrice , ".2f") , sep ="")
print ("Percent Off: " ,format( percentOff , "d") ,"%" , sep ="")
if ( percentOff >= 50):
   print ("You got a great sale!")
   print ("Congratulations!")
print ("Done!")
```

Question 17: 7 points

Enter and execute the following code.

```
tempString = input("Enter the water temperature in degrees Fahrenheit: ")
temperature = int ( tempString )
if temperature >= 212:
    print("Water is boiling.")
else :
    print("The water is not boiling.")
```

(a) (3 points) Test the program at least three times. List the three test inputs you used and the corresponding output. Be sure you test each part of the condition.

Input: 32 Output: "The water is not boiling."

Input: 212 Output: "Water is boiling."

Input : 250 Output: "Water is boiling."

(b) (2 points) Explain why the data you chose were the best data to use to thoroughly test for the program.

The first value I tested, 32, makes the condition `temperature >= 212` false which means that statements in the `else` block should be executed.
The condition `temperature >= 212` is true for all values of temperature that are greater than or equal to 212. So, I used 250 to test the "greater than" part of the condition, and 212 to test the "or equal to" part of the condition. These two values should cause statements in the `if` block to be executed.

(c) (2 points) Now add another print statement to the Python program above so that it prints "That's really hot!" when the water is 212 degrees or hotter. Rewrite the if/else below with this statement included.

```
if temperature >= 212:
    print("Water is boiling.")
    print("That's really hot!")
else :
    print("The water is not boiling.")
```

Question 18: 1 point

Suppose you want to determine if a student is ready to graduate. The three criteria for graduation are that the student has earned at least 120 credits, their major GPA is at least 2.0 and their general GPA is also at least 2.0.

(a) (1 point) Select the correct Boolean Expression to determine if a student is ready to graduate.

- ○ `numCredits >= 120 or majorGPA >= 2.0 or overallGPA >= 2.0`
- ○ `numCredits > 120 and majorGPA > 2.0 or overallGPA > 2.0`
- ○ `numCredits > 119 and majorGPA >= 2.0 and overallGPA >= 2.0`
- ○ `numCredits >= 120 and majorGPA >= 2.0 and overallGPA >= 2.0`

Question 19: 8 points

We want to test the expression from Question 18.

(a) (2 points) Pick two values for **`numCredits`**. One that will make its comparison `True`. One that will make it `False`.
225 (True) and 103 (False)

(b) (2 points) Pick two values for **`majorGPA`**. One that will make its comparison `True`. One that will make it `False`.
4.0 (True) and 1.50 (False)

(c) (2 points) Pick two values for **`overallGPA`**. One that will make its comparison `True`. One that will make it `False`.
4.0 (True) and 1.50 (False)

(d) (2 points) If we want to test all possible combinations of the expression, how many tests do we need to do?
9 Tests

Question 20: 4 points

Write the code for an **`if`** statement that adds 5 to the variable **`num1`** if the value stored in the variable **`testA`** equals 25. Otherwise subtract 5 from **`num1`**.

```
3   # Get variable values
4   testA = int(input('Enter value for testA: '))
5   num1 = int(input('Enter value for num1: '))
6
7   # Decison Structure:
8   if testA == 25:
9       num1 += 5
10  else:
11      num1 -= 5
```

Question 21: 4 points

Write a Python program that prompts the user for a word. If the word comes between the words **apple** and **pear** alphabetically, print a message that tells the user that the word is valid, otherwise, tell the user the word is out of range.

```
1   # Author: Tony Kabilan Okeke
2   # Date: 01.27.2022
3
4   # Collect input
5   word = input('Please provide a word: ')
6
7   # Return Output
8   if ('apple' < word) and ('pear' > word):
9       print(f'"{word}" is valid.')
10  else:
11      print(f'"{word}" is out of range.')
```

```
[py-env][kabil] >> ~/../cs171/Wk4 python lab4_Q21.py
Please provide a word: cat
"cat" is valid.
[py-env][kabil] >> ~/../cs171/Wk4 python lab4_Q21.py
Please provide a word: alien
"alien" is out of range.
[py-env][kabil] >> ~/../cs171/Wk4 python lab4_Q21.py
Please provide a word: zebra
"zebra" is out of range.
```

Question 22: 4 points

Write a Python program that prompts the user for the cost of two items to be purchased. Then prompt the user for payment. If the amount entered is less than the total cost of the two items, print a message that states how much is still owed. Otherwise, print a thank you message and state how much change will be given.

```python
# Author: Tony Kabilan Okeke
# Date: 01.27.2022

# Collect inputs
item1 = float(input( "Enter the price of the first item: " ))
item2 = float(input( "Enter the price of the second item: " ))
payment = float(input( "How much did you pay? " ))

# Compute total cost
total_cost = item1 + item2

# Print output
if ( payment < total_cost ):
    balance = total_cost - payment
    print (f"You still owe ${balance:.2f}")
else:
    change = payment - total_cost
    print (f"Your change is ${change:.2f}.\nThank you!")
```

```
[py-env][kabil] ❯❯ ~/../cs171/Wk4 python lab4_Q22.py
Enter the price of the first item: 10
Enter the price of the second item: 10
How much did you pay? 20
Your change is $0.00.
Thank you!
[py-env][kabil] ❯❯ ~/../cs171/Wk4 python lab4_Q22.py
Enter the price of the first item: 10
Enter the price of the second item: 10
How much did you pay? 9
You still owe $11.00
[py-env][kabil] ❯❯ ~/../cs171/Wk4 python lab4_Q22.py
Enter the price of the first item: 10
Enter the price of the second item: 10
How much did you pay? 30
Your change is $10.00.
Thank you!
```

Question 23: 4 points

Write a Python program that prompts the user for a multiple of 5 between 1 and 100. Print a message telling the user whether the number they entered is valid.

```python
# Author: Tony Kabilan Okeke
# Date: 01.27.2022

# Collect input
num = int(input("Please Enter a Multiple of 5 Between 1 and 100: "))

# Print Output
if (num % 5 == 0) and (num >= 1) and (num <= 100):
    print(f"{num} is valid.")
else:
    print(f"{num} is invalid.")
```

```
[py-env][kabil] ❯❯ ~/../cs171/Wk4 python lab4_Q23.py
Please Enter a Multiple of 5 Between 1 and 100: 50
50 is valid.
[py-env][kabil] ❯❯ ~/../cs171/Wk4 python lab4_Q23.py
Please Enter a Multiple of 5 Between 1 and 100: 23
23 is invalid.
[py-env][kabil] ❯❯ ~/../cs171/Wk4 python lab4_Q23.py
Please Enter a Multiple of 5 Between 1 and 100: 123
123 is invalid.
```