

Name: \_\_\_\_\_

Date: \_\_\_\_/\_\_\_\_/\_\_\_\_

## Midterm #1 - Sample Questions

1. Write out the result of this code. If there's output, write out what the code will output. If there's an error, write out why an error happened and how to fix it. (2 points)

```
distance = 150
time = 10
velocity = distance / time
print("The velocity is " + velocity)
```

The result is an error: adding two types that cannot be added - a float and a string.

2. What is the resulting Boolean value for the following expressions? (4 points)

- |   |                 |
|---|-----------------|
| a) False or True and False                                | a) <u>False</u> |
| b) True and not (9 < 5)                                   | b) <u>True</u>  |
| c) 24 / 12 >= 2 and "hello" != "goodbye"                  | c) <u>True</u>  |
| d) answer = "yeah"<br>answer == "yes" or answer == "yeah" | d) <u>True</u>  |

3. Some tricky ones. What is the resulting Boolean value for the following expressions? If there's an error, write error. (3 points)

- |                  |                 |
|------------------|-----------------|
| a) 55 != "hello" | a) <u>True</u>  |
| b) "2" == 2      | b) <u>False</u> |
| c) "hello" > 35  | d) <u>Error</u> |

4. 00001001 is 9 in decimal, and 38 is 00100110 in binary. (1 point)

5. What's the difference between a compiled and an interpreted language? (1 point)

A compiled language requires an explicit compilation step.

(When a language is compiled, it means that it has to be translated into a form that the computer can understand by using a program called a compiler. An interpreted language, on the other hand, without the programmer having to run a separate translation process – it is done behind the scenes by the interpreter).

6. What is the output of this code if the user enters 5 (2 points)

```
def sum_numbers_up_to(last_number):
    total = 0
    for number in range(1, last_number + 1):
        total += number
    print(total)

def main():
    n = int(input('Number please!\n'))
    sum_numbers_up_to(n)

main()
```

The output is 15

7. Create a **while** loop that prints out "Hi, how are you?" **forever**. (2 points)

Example Output:

```
Hi, how are you?      while True:
.                      print('Hi, how are you?')
.
Hi, how are you?
```

8. You've just been hired by a party planning partnership called Party People Plus (whew, that's a lot of p's). They would like you to **write a program that prints out numbers in a count down based on a value entered by the program's user**. (6 points)

The program should do the following:

- ask the user for number (and store the number entered) by saying: "How long before the party!?"
- print out: "Here's the count down!"
- count down from the number entered to 1 by printing out each number
- if the current number in the countdown is less than or equal to three, surround the number with asterisks
- at the end of the countdown, print out: "Party Time!!!"
- Example Output:

```
How long before the party!?
> 5
Here's the count down!
5
4
*3*
*2*
*1*
Party Time!!!
```

```
party_time = int(input("How long before the party!?\n> "))
for number in range(party_time, 0, -1):
    if number > 3:
        print(number)
    else:
        print("*" + str(number) + "*")
```

9. Write a program that prints out a daily budget for a vacation. (7 points)

- ask the user for the number of days** of their vacation, as well as their **destination**
- based on the number of days the vacation is, **ask for how much they're spending on each day** (x is the day's number):  
How much are you spending on day [x]?
- once you're done, **print out a report** with the **destination**, **total cost**, and **amount** per day
- hint: how can the printing of the report be delayed until the end of the program?

Example Output:

```
How many days is your vacation?
> 2
Where are you going?
> Ottawa
How much are you spending on day 1?
> 100
How much are you spending on day 2?
> 200
Your vacation to Ottawa is $400
day 1 - $100
day 2 - $200
```

```
days = int(input("How many days is your vacation?\n> "))
where = input("Where are you going?\n> ")
report = ''
total = 0
for day in range(1, days + 1):
    amt = int(input('How much are you spending on day ' + str(day) + '\n> '))
    total += amt
    report += 'day ' + str(day) + ' - $' + str(amt) + '\n'
print('Your vacation to ' + where + ' is ' + str(total))
print(report)
```

10. Define the following terms (1 and 1/2 points)

**program:** a sequence of instructions that specifies to a computer actions and computations to be performed

**function:** a named sequence of statements that perform a specific task or a useful operation. Functions can optionally have inputs (parameters) and outputs (return value(s)).

**call (as in call a function):** execute or *run* a function. To call a function add parentheses after the function name, and within those parentheses optionally add a comma separated list of arguments (leave parentheses empty if no arguments).

11. Answer the questions in the right column about the condition snippet of code in the left column. (6 points)

Code	Question 1	Question 2
<pre>answer = input("Type something plz: ") while answer != "yes" and answer != "yeah":     print("I'm in yr while loop")     answer = input("Type something plz: ")</pre>	<p>If a user types in "no", what – if anything – would this program output?</p> <p>I'm in yr while loop</p>	<p>Write down all of the values that a user could input to prevent the while loop from running.</p> <p>yes or yeah</p>
<pre>answer = input("Type something plz: ") another = input("Type another thing plz: ") if answer == "no" and another == "n":     print("Ohhh no!") elif answer == "yes" or another == "y":     print("Yeaaahhhh") else:     print("Huh?")</pre>	<p>If a user types in the word "yes" for the first question and "comfy-pants" for the second, what – if anything – would this program output?</p> <p>Yeaaahhhh</p>	<p>What values must the user enter for this program to print "Ohhh no!"?</p> <p>answer: no another: n</p>
<pre>def greet(name, num_exclamation_points):     punctuation = '!' * num_exclamation_points     print('Hi ' + name + punctuation)</pre>	<p>How many parameters does the function, greet, take?</p> <p>2</p>	<p>How would you call this function so that the output is "Hi Harriet!!!"?</p> <p>greet('Harriett', 3)</p>

12. Write a program that **continually asks for numbers until the user enters three consecutive numbers that are all the same**. Once three consecutive numbers that are equal are entered, the program will **print out 'Done!'**. Assume that the user only enters numbers.

Hints:

- you will have to keep track of a previous value (but how will you initialize it?)
- 3 consecutive numbers means that the 1<sup>st</sup> in the series wasn't equal to a previous, but the next two are

Example Output:

```
Please enter a number
> 5
Please enter a number
> 5
Please enter a number
> 7
Please enter a number
> 7
Please enter a number
> 7
Please enter a number
> 7
Done!
```

```
consecutive = 0
prev = ''
while consecutive < 2:
    num = int(input('Please enter a number\n> '))
    if prev == num:
        consecutive += 1
    else:
        consecutive = 0
    prev = num
```

13. Here is a partial implementation of a program that:

- a) **generates a series of numbers, each a random value** from 1 through 11
- b) **prints out** each generated number
- c) **keeps a running total of the generated numbers**
- d) **stops printing** the numbers when the **sum** of the numbers generated reaches **21 or over**

Fill in the blanks to complete the implementation. Example output is provided on the left. (5 points)

<u>Example Run 1:</u>	<u>Example Run 2:</u>	<code>import random</code>
7	2	<code>total = 0</code>
10	8	<code>while total &lt; 21:</code>
5	10	<code>    random_value = random.randint(1, 11)</code>
	6	<code>    print(random_value)</code>
		<code>    total = total + random_value</code>

14. This chart describes the order in which types of operators are evaluated, along with examples of each. Fill in the blanks: (4 points)

Order Evaluated (1, 2, 3 or 4)	Type of Operator	Example Operator
3	Comparison	>, <, >=, <=, ==, !=
4	Logical	not, and, or
1	Parentheses	()
2	Numeric and/or String Operators	+, *, -, /

15. Use DeMorgan's law to write an equivalent if statement adjacent to the one shown below (1 point):

<code>if not (age &gt;= 21 and has_id == True):</code>	<code>if __not (age &gt;= 21) or not (has_id == True)__:</code>
<code>    print("Not allowed in the pub")</code>	<code>    print("Not allowed in the pub")</code>

Use logical opposites to remove nots from you answer above (1 point):

```
if __age < 21 or has_id != True__:
    print("Not allowed in the pub")
```

16. Define a function called `calculate_volume`. It should have three parameters, `length`, `width`, and `height`. The function should multiply all 3 values and print out the result. Define a second function called `main`. This function should call the first one using values collected from user input. Finally, call `main`. The output of the final program should be something like (5 points):

What is the object's length?

> 5

What is the object's width?

> 2

What is the object's height?

> 3

The objects volume is 30.

```
def calculate_volume(length, width, height):
    print("The object's volume is " + str(length * width * height))

def main():
    l = int(input("What is the object's length?\n> "))
    w = int(input("What is the object's width?\n> "))
    h = int(input("What is the object's height?\n> "))

    calculate_volume(l, w, h)
main()
```