

APS106

MIDTERM REVIEW JEOPARDY

RULES OF THE GAME

- Form a team with your table or tables around you (sizes ~2-4)
- Choose a team name for the scoreboard
- Go to <https://www.multibuzz.app>
- Enter the Room Code TQCWMG and your Team Name
- If you get the answer right, you gain the points, and pick the next category
- If you get the answer wrong, you lose the points, and the next person who buzzed in is up
- Ask questions if you're confused! Remember, if you're confused, so are others!

PANEL

THE GOOD, THE BAD, AND THE FUNCTION	FILE CLUB	HOW TO TRAIN YOUR WHILE LOOP	TO OUTPUT OR NOT TO OUTPUT	THE HITCHHIKER'S GUIDE TO PYTHON
\$250	\$250	\$250	\$250	\$250
\$500	\$500	\$500	\$500	\$500
\$750	\$750	\$750	\$750	\$750
\$1000	\$1000	\$1000	\$1000	\$1000

THE GOOD, THE BAD, AND THE FUNCTION · \$250

What is the result?

```
def power3(num):  
    print(num**3)
```

```
result1 = power3(5)  
result2 = result1 + 1
```

What is the outcome of executing the code above?

- result2 refers to the value 126
- result2 refers to the value 126 and is printed
- result1 refers to the value 125 and result2 refers to the value 126
- TypeError

[Question board](#)

THE GOOD, THE BAD, AND THE FUNCTION · \$250

```
def power3(num):  
    print(num**3)
```

```
result1 = power3(5)  
result2 = result1 + 1
```

What is the outcome of executing the code above?

- result2 refers to the value 126
- result2 refers to the value 126 and is printed
- result1 refers to the value 125 and result2 refers to the value 126
- TypeError

 BACK TO PANEL

Question board

THE GOOD, THE BAD, AND THE FUNCTION · \$500

What is the output?

```
def howbig(n):
    if n > 190:
        return "It's huge."
    elif n > 160:
        return "It's pretty big."
    else:
        return "It's not so big."
print(howbig(195))
```

 BACK TO PANEL

Question board

THE GOOD, THE BAD, AND THE FUNCTION · \$500

```
def howbig(n):
    if n > 190:
        return "It's huge."
    elif n > 160:
        return "It's pretty big."
    else:
        return "It's not so big."
print(howbig(195))
```

It's huge.

It's huge.

It's pretty big.

It's huge.

It's pretty big.

It's pretty big.

It's not so big.

◀ BACK TO PANEL

[Question board](#)

THE GOOD, THE BAD, AND THE FUNCTION · \$750

```
def test_function(a_boolean_variable):  
  
    if a_boolean_variable != True:  
        return True  
    else:  
        return False
```

Which of the following statements can replace the body of `test_function` and produce the same result?

- (a) `return False`
- (b) `return a_boolean_variable`
- (c) `return True`
- (d) `return not a_boolean_variable`
- (e) None of the above

 BACK TO PANEL

[Question board](#)

THE GOOD, THE BAD, AND THE FUNCTION · \$750

```
def test_function(a_boolean_variable):  
  
    if a_boolean_variable != True:  
        return True  
    else:  
        return False
```

Which of the following statements can replace the body of `test_function` and produce the same result?

- (a) `return False`
- (b) `return a_boolean_variable`
- (c) `return True`
- (d) `return not a_boolean_variable`
- (e) `None of the above`

 BACK TO PANEL

[Question board](#)

THE GOOD, THE BAD, AND THE FUNCTION - \$1000

How many times in total is the function `f()` called?

```
def f():
    return False

x = 2
if f() and x > 3:
    print("Blue pill")
elif x <= 2 or f():
    print("Red pill")
    if 2 + x > 4 and f():
        f()
```

- (a) 5
- (b) 4
- (c) 3
- (d) 2
- (e) 1

 BACK TO PANEL

Question board

THE GOOD, THE BAD, AND THE FUNCTION - \$1000

```
def f():
    return False

x = 2
if f() and x > 3:
    print("Blue pill")
elif x <= 2 or f():
    print("Red pill")
    if 2 + x > 4 and f():
        f()
```

- (a) 5
- (b) 4
- (c) 3
- (d) 2
- (e) 1

 BACK TO PANEL

[Question board](#)

FILE CLUB · \$250

What is the output?

```
string_a = 'what a lovely day today! '
string_a[5:-5]
```

◀ BACK TO PANEL

[Question board](#)

FILE CLUB · \$250

```
string_a = 'what a lovely day today!'
string_a[5:-5]
```

```
In [1]: string_a = "What a lovely day today!"
        string_a[5:-5]
```

```
Out[1]: 'a lovely day t'
```

 BACK TO PANEL

[Question board](#)

FILE CLUB · \$500

What is the result in the f.txt file?

```
f = open('f.txt', 'w')
f.write('Hello\n')
f.write('\n')
f.close()
f = open('f.txt', 'w')
f.write('Fun! ')
f.close()
```

- a) Fun!
- b) 'Hello'
 'Fun!'
- c) Hello
 Fun!
- d) 'Hello Fun!'

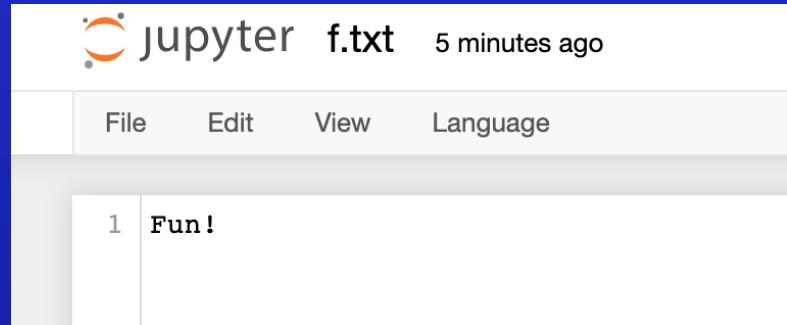
 BACK TO PANEL

[Question board](#)

FILE CLUB · \$500

What is the result in the f.txt file?

```
f = open('f.txt', 'w')
f.write('Hello\n')
f.write('\n')
f.close()
f = open('f.txt', 'w')
f.write('Fun! ')
f.close()
```



The screenshot shows a Jupyter Notebook interface with a toolbar at the top labeled "jupyter f.txt 5 minutes ago". Below the toolbar is a menu bar with "File", "Edit", "View", and "Language". The main area displays a single cell with the output "1 Fun!".

- a) Fun!

BACK TO PANEL

[Question board](#)

FILE CLUB · \$750

How do you print out the first two characters uppercase per item?

jupyter groceries.txt a minute ago

File Edit View Language

```
1 apples
2 bananas
3 kiwi
```



```
grocery_list = open("groceries.txt", "r")
for item in grocery_list:
    ...
    ...
    grocery_list.close()
```

```
AP
BA
KI
```

BACK TO PANEL

Question board

FILE CLUB · \$750

How do you print out the first two characters uppercase per item?

jupyter groceries.txt a minute ago

File Edit View Language

```
1 apples
2 bananas
3 kiwi
```



```
grocery_list = open("groceries.txt", "r")
for item in grocery_list:

    item = item.upper()
    print(item[0:2])

grocery_list.close()
```

AP

BA

KI

BACK TO PANEL

[Question board](#)

FILE CLUB · \$1000

What is the output?



jupyterhub test.txt a few seconds ago

File Edit View Language

```
1 APS111, 04, 20, 2017, 9:30
2 APS111, 04, 20, 2017, 2:00
3 APS164, 04, 18, 2017, 9:30
```

```
: myfile = open('test.txt', 'r')
for row in myfile:
    print(row[0:6] + ' exam on ' + row[12:14] + 'th at ' + row[22:28])
myfile.close()
```

BACK TO PANEL

[Question board](#)

FILE CLUB · \$1000

jupyter test.csv 2 minutes ago

	File	Edit	View	Language
1	APS111, 04, 20, 2017, 9:30			
2	APS111, 04, 20, 2017, 2:00			
3	APS164, 04, 18, 2017, 9:30			

```
myfile = open('test.txt', 'r')
for row in myfile:
    print(row[0:6] + ' exam on ' + row[12:14] + 'th at ' + row[22:28])
myfile.close()
```

APS111 exam on 20th at 9:30

APS111 exam on 20th at 2:00

APS164 exam on 18th at 9:30

BACK TO PANEL

Question board

HOW TO TRAIN YOUR WHILE LOOP · \$250

What are the values of `var1` and `var2` after the following code segment is executed and the while loop finishes?

```
var1 = 0;
var2 = 2;

while ((var2!=0) and ((var1 / var2) >= 0)):

    var1 = var1 + 1;
    var2 = var2 - 1;
```

◀ BACK TO PANEL

[Question board](#)

HOW TO TRAIN YOUR WHILE LOOP · \$250

What are the values of `var1` and `var2` after the following code segment is executed and the while loop finishes?

```
In 8  1 var1 = 0;
      2 var2 = 2;
      3
      4 while((var2!=0) and ((var1 / var2) >= 0)):
      5     var1 = var1 + 1
      6     var2 = var2 - 1
      7
      8 print (var1, var2)
```

2 0

BACK TO PANEL

Question board

HOW TO TRAIN YOUR WHILE LOOP · \$500

What will this code print?

```
n = 10
answer = 1
while n > 0:
    answer = answer + n
    n = n + 1
print(answer)
```

 BACK TO PANEL

Question board

HOW TO TRAIN YOUR WHILE LOOP · \$500

What will this code print?

```
n = 10
answer = 1
while n > 0:
    answer = answer + n
    n = n + 1
print(answer)
```

Nothing! Infinite loop!

◀ BACK TO PANEL

[Question board](#)

HOW TO TRAIN YOUR WHILE LOOP · \$750

What will the code print?

```
i = 0
j = 6

while 0 < j < 10:

    print(i, end = " ")

    if j % 3 == 0:
        j = j // 2
    elif j% 2 == 0:
        j = j * 2
    else:
        j = j + 1

    i += 1
print(j)|
```

⬅ BACK TO PANEL

[Question board](#)

HOW TO TRAIN YOUR WHILE LOOP · \$750

What will the code print?

```
i = 0
j = 6

while 0 < j < 10:

    print(i, end = " ")

    if j % 3 == 0:
        j = j // 2
    elif j% 2 == 0:
        j = j * 2
    else:
        j = j + 1

    i += 1
    print(j)|
```

0	3
1	1
2	2
3	4
4	8
5	16

◀ BACK TO PANEL

Question board

HOW TO TRAIN YOUR WHILE LOOP · \$1000

- c) In the code below, how many times is the function `f` executed?

```
def f(n):
    return n - 3

x = 4
while x > 0:
    x = f(x)
```

 BACK TO PANEL

Question board

HOW TO TRAIN YOUR WHILE LOOP · \$1000

- c) In the code below, how many times is the function `f` executed?

```
def f(n):
    return n - 3

x = 4
while x > 0:
    x = f(x)
```

2

◀ BACK TO PANEL

Question board

TO OUTPUT OR NOT TO OUTPUT · \$250

What is the output?

```
x = 100  
y = 50  
print(x and y)
```

◀ BACK TO PANEL

Question board

TO OUTPUT OR NOT TO OUTPUT · \$250

What is the output?

```
In 11  1 x = 100  
      2 y = 50  
      3 print (x and y)
```

50

⬅ BACK TO PANEL

[Question board](#)

TO OUTPUT OR NOT TO OUTPUT · \$500

What is the output?

```
x = 10
y = 50
if x ** 2 > 100 and y < 100:
    print(x, y)
```

◀ BACK TO PANEL

Question board

TO OUTPUT OR NOT TO OUTPUT · \$500

What is the output?

```
x = 10
y = 50
if x ** 2 > 100 and y < 100:
    print(x, y)
```

Nothing!

◀ BACK TO PANEL

[Question board](#)

TO OUTPUT OR NOT TO OUTPUT · \$750

Q6. What is the output of the following commands?

```
>>> counter = 1  
>>> time = 60 * counter  
>>> counter += 1  
>>> time = 60 * counter  
>>> time
```

 BACK TO PANEL

Question board

TO OUTPUT OR NOT TO OUTPUT · \$750

Q6. What is the output of the following commands?

```
>>> counter = 1  
>>> time = 60 * counter  
>>> counter += 1  
>>> time = 60 * counter  
>>> time
```

120

◀ BACK TO PANEL

[Question board](#)

TO OUTPUT OR NOT TO OUTPUT · \$1000

Which of the following evaluates to True when a is equal to b or when a is equal to 5?

- A. $a == b == 5$
- B. $a = b$ or $a = 5$
- C. $a == b$ or $a == 5$
- D. $a = b$ and $a = 5$
- E. $a == b$ and $a = 5$

 BACK TO PANEL

Question board

TO OUTPUT OR NOT TO OUTPUT · \$1000

Which of the following evaluates to True when a is equal to b or when a is equal to 5?

- A. $a == b == 5$
- B. $a = b$ or $a = 5$
- C. $a == b$ or $a == 5$
- D. $a = b$ and $a = 5$
- E. $a == b$ and $a = 5$

 BACK TO PANEL

[Question board](#)

THE HITCHHIKER'S GUIDE TO PYTHON · \$250

Q1. Once the following commands have been executed in the shell, what is the value of area?

```
>>> length = 3  
>>> height = 2  
>>> area = length * height  
>>> height = 4  
>>> area
```

 BACK TO PANEL

[Question board](#)

THE HITCHHIKER'S GUIDE TO PYTHON · \$500

c) [1 mark] In the code below, how many times does the body of the while-loop get executed?

```
x = 0
while x < 10:
    x *= 2
    x -= 1
```

 BACK TO PANEL

Question board

THE HITCHHIKER'S GUIDE TO PYTHON · \$500

c) [1 mark] In the code below, how many times does the body of the while-loop get executed?

```
x = 0
while x < 10:
    x *= 2
    x -= 1
```

- (a) 10
- (b) Infinity (it's an infinite loop)
- (c) 5
- (d) 6
- (e) 0

 BACK TO PANEL

[Question board](#)

THE HITCHHIKER'S GUIDE TO PYTHON · \$750

Q4. What goes in the blank in the following code to so that the output is 7.0?

```
>>> ____ + 3 ** 2 // 2  
7.0
```

- a. $-(7 \% 4)$
- b. $2 ** 3$
- c. $6 / 2$
- d. 3
- e. 6.0

 BACK TO PANEL

[Question board](#)

THE HITCHHIKER'S GUIDE TO PYTHON · \$750

Q4. What goes in the blank in the following code to so that the output is 7.0?

```
>>> ____ + 3 ** 2 // 2  
7.0
```

- a. $-(7 \% 4)$
- b. $2^{**} 3$
- c. $6 / 2$
- d. 3
- e. 6.0

 BACK TO PANEL

[Question board](#)

THE HITCHHIKER'S GUIDE TO PYTHON · \$1000

Q12. What is the output of the execution of the following code snippets.

```
def f(n):
    return 3 - n
def g(n):
    return 5 * n
def h(n):
    return 2 * n
def doto(value, func):
    if func == "f":
        return f(value)
    elif func == "g":
        return g(value)
    return h(value)
print(doto(doto(2, "g"), "r") + doto(2, "f"))
```

 BACK TO PANEL

Question board

THE HITCHHIKER'S GUIDE TO PYTHON · \$1000

Q12. What is the output of the execution of the following code snippets.

```
def f(n):
    return 3 - n
def g(n):
    return 5 * n
def h(n):
    return 2 * n
def doto(value, func):
    if func == "f":
        return f(value)
    elif func == "g":
        return g(value)
    return h(value)
print(doto(doto(2, "g"), "r") + doto(2, "f"))
```

21

 BACK TO PANEL

[Question board](#)

FINAL JEOPARDY

QUESTION 1 · FINAL JEOPARDY

Question 3. [12 marks]

a) [4 marks] The following *poorly written* Python function is intended to identify if an integer between 5 and 12 is even or odd and divisible by three or four. It should then print those properties to the screen. There are 4 errors in the code. Circle each error and fill in the table on the right-hand side, indicating line number and writing the complete correct line of code.

```
1 def what_factors(num):
2     """
3         (int) -> NoneType
4     """
5     if num == 5:
6         print("Odd.")
7     if num == 6:
8         print("Even.")
9         print("Divisible by 3.")
10    if num == 7:
11        print("Odd.")
12    if num == 8:
13        print("Even.")
14        print("Divisible by 4.")
15    if num == 8:
16        print("Odd.")
17        print("Divisible by 3.")
18    if num == 10:
19        print("Even.")
20    if num == 11:
21        print("Odd.")
22    else num == 12:
23        print("Even.")
24        print("Divisible by 3.")
25        print("Divisible by 4.")
```

Line #	Correct Code

 BACK TO PANEL

QUESTION 1: FINAL JEOPARDY

```
1 def what_factors(num):
2     """
3         (int) -> NoneType
4     """
5     if num == 5:
6         print("Odd.")
7     if num == 6:
8         print("Even.")
9         print("Divisible by 3.")
10    if num == 7:
11        print("Odd.")
12    if num == 8:
13        print("Even.")
14        print("Divisible by 4.")
15    if num == 8:
16        print("Odd.")
17        print("Divisible by 3.")
18    if num == 10:
19        print("Even.")
20    if num == 11:
21        print("Odd.")
22    else num == 12:
23        print("Even.")
24        print("Divisible by 3.")
25        print("Divisible by 4.")
```

There are 5 errors (line 2 is arguable) and so the students should identify any 4 of the 5.

4 Marks Total - 0.5 for error, 0.5 for the correction.

Line #	Correct Code
2	"""
5	if num == 5:
11	print("Odd.")
15	if num == 9:
22	else: also acceptable: elif num == 12: if num == 12:

 BACK TO PANEL

QUESTION 2 · FINAL JEOPARDY

Write a function that evaluates $\tanh^{-1}(x)$ from the series

$$\tanh^{-1}(x) = x + x^3/3 + x^5/5 + x^7/7 + x^9/9 + \dots$$

where $x < 1.0$ and $x > -1.0$. Notice that each successive term in the series is smaller than the previous one.

The function should take x and epsilon, a small number. Your function should return the estimate of $\tanh^{-1}(x)$ once the change in the estimate from adding a new term would be less than or equal to epsilon. Do not use the math module.

If x is not between -1.0 and 1.0, the program should repeatedly ask for input until it is.

When the input is valid, the program should call your function to estimate $\tanh^{-1}(x)$. Your function returns the approximation and then your code prints the value of $\tanh^{-1}(x)$.

 BACK TO PANEL

QUESTION 2 · FINAL JEOPARDY

```
def estimate_tanh_inv(x, epsilon):
    '''(float, int) -> float
    Returns an estimate of tanh^-1 based on the series:
    tanh-1(x) = x + x^3/3 + x^5/5 + x^7/7 + x^9/9 + ...
    Stops when the estimate is within next item in the series is less than
    or equal to epsilon
    '''
    tanh_inv = 0
    term = x
    num = 3
    while abs(term) > epsilon:
        #print(term, num)
        tanh_inv += term
        term = x**num/num
        num += 2

    return tanh_inv

x = -2
while x < -1.0 or x > 1.0:
    x = float(input("Enter x between -1 and 1: "))
    eps = float(input("Enter epsilon: "))

tanh_inv = estimate_tanh_inv(x, eps)
print("tanh-1(", x, ") = ", tanh_inv, sep="")
```

 BACK TO PANEL