

## CIS 30A Midterm Exam

Name Aamir Khan

You can use any resources in the Canvas and your textbook but do not Google or use Chegg or any other internet resources. Do not discuss programming assignment questions with anyone including your classmates. If you have any questions about the programming assignment, contact your instructor. Do not discuss programming assignment questions in Discussion Board.

Please submit your work to Canvas by 11:59 PM 10/16/2022. Be sure to submit your work as single pdf file along with signed signature page

**NO LATE ASSIGNMENT WILL BE ACCEPTED.**

"I affirm that I will not give or receive any unauthorized help on this exam, and that all work will be my own."

Signature Aamir Khan



Q1. (3 pts) Explain the difference between a *variable* and an *expression*.

Answered on the back.

Q2. (2 pts) What does *immutable* mean in Python? Include an example.

Answered on the back.

Q3. (4 pts) Circle the legal identifiers. Briefly indicate why the illegal ones are illegal.

pOiNts

#\_of\_points → doesn't start with a letter or underscore.  
2ndPoint → starts with a number.  
pointx+y → has a +. Only \_ are allowed as special characters.

point2

x value

this point → has a space. whitespace characters aren't allowed.  
point\_Two

Q4. (4 pts) Write a statement that prints the line She yelled, "Run for the Hills!"

print("She yelled, \"Run for the Hills!\"")

Q5. (2 pts) Compute the values using Python operation

(a)  $2^{**}5 = 2^5 = 32$

(b)  $26 // 7 = 3$  (Floor division)

(c)  $34 \% 5 = 4$  (Modulus or remainder)  
 $34 / 5 = 6.8$

(d)  $19 / 5 = 3.8$

Q6. (4 pts) Explain why Python can hold integers exactly but not floats.

Answered on the back.

Q7. (4 pts) If  $s = \text{"Albatross!"}$ , what is printed (or error)?

print("1:", s[5]) 1: r

print("2:", s[-3]) 2: s

print("3:", s[4:8]) 3: tros

print("4:", s[:7]) 4: Albatro

print("5:", s[1:]) 5: lbatross!

print("6:", len(s)) 6: 10

$s = \text{"Albatross!"}$   
0 1 2 3 4 5 6 7 8 9 → Index

Q8. (2 pts) Circle all legal Python string operation.

(a) "123" + "abc"

(b) "ABC"\*3

(c) "123" - 3

(d) "ABC".lower()

Everything except c.

Q9. (2 pts) True or False: In Python  $4 + 5$  and  $4.0 + 5.0$  produce the same result. Explain your answer.

False, although they may appear to be the same.

Floats are approximate, so the result isn't exactly the same.



Q1. Explain the difference between a variable and an expression.

A variable is a container for information. It is an identifier for some data. On the other hand, an expression is a combination of variables and operators. Expressions may also include literals.

Q2. What does immutable mean in Python? Include an example.

Immutable means that something can't be changed or modified. Strings are immutable in Python, so they can't be changed once they're assigned. If you try to change a string, you'll get an error.

Q6. Explain why Python can hold integers exactly but not floats.

Since integers can be represented exactly in binary, Python stores them exactly. However, floats can't be represented exactly in binary, so Python stores them approximately.



Q10. (2 pts) Circle all the correct statement(s).

- (a) Slicing can only be applied to strings, not lists *False.*
- ☒ (b) A string can be used in creating another string.
- (c) Once created, lists cannot be modified. *False.*
- ☒ (d) Strings are a sequence of only characters; lists are a sequence of any types or combination of types.

Q11. (5 pts) Write Python code for following statements.

*Answered on the back.*

- a) Ask user to enter first number and assign it to num1
- b) Ask user to enter second number and assign it to num2
- c) Add these two values and assign it to sum
- d) print the output as The sum of num1 and num2 is sum using format method. `"".format()`

Q12. (4 pts) If I do `t = "Bread"`, followed by `t[4]="k"` what happens? Briefly explain. *You get an error because you're trying to change a string, which is immutable.*

Q13. (2 pts) Circle all operations that will produce the last element of a string `mystring`?

- ☒ (a) `mystring[len(str) - 1]`
- (b) `mystring.(len(str-1))` *Invalid.*
- (c) `mystring[len(str)]` *Invalid.*
- ☒ (d) `mystring[-1]`



Q11. Write Python code for the following statements.

```
num1 = float(input("Enter your first number: "))
```

```
num2 = float(input("Enter your second number: "))
```

```
sum = num1 + num2
```

```
print("The sum of {0} and {1} is {2}").
```

```
format(num1, num2, sum))
```



Q14. (4 pts) Consider following code. What list output would you expect after print command?

```
fruits1 = ['apple', 'orange', 'mango', 'banana', 'watermelon']  
fruits1[0] = 'strawberry'
```

1 → print(fruits1)

```
fruits1.insert(0, 'peach')
```

2 → print(fruits1)

*Answered on  
the back*

Q15. (2 points) What does IF statement use indentation to tell Python?

*Answered on the back.*

Q16 (4 points) Simplify the following segment, taking out any unnecessary comparisons.

```
if age > 68:
```

```
    print("Retire-able")
```

```
if age <= 22:
```

```
    print("Educate-able")
```

```
if (age > 22 and age <= 68):
```

```
    print("Employ-able")
```

```
if age <= 22:
```

```
    print("Educate-able")
```

```
elif (age > 22 and age <= 68):
```

```
    print("Employ-able")
```

```
else:
```

```
    print("Retire-able")
```



Q14. what list output would you expect after each `print()` command.

After 1<sup>st</sup> print:

```
['strawberry', 'orange', 'mango', 'banana', 'watermelon']
```

After 2<sup>nd</sup> print:

```
['peach', 'strawberry', 'orange', 'mango', 'banana', 'watermelon']
```

Q15. If-statements use indentation to tell Python what?

If-statements use indentation to tell Python what code belongs inside of the if-statement code block. If code is indented, Python will know that it is a part of the code block. All code that is inside of the code block will execute if the statement is true.