1. Strings are **collection data types**, what does that mean?

2. What is the **ordinal value** of a character?

3. Consider the following statement:

   ```python
   s = "Call Me Maybe"
   ```

   Which of the following changes `s` to refer to "Call Me Perhaps"?

   1. `s[-5:] = 'Perhaps'`
   2. `s = s[: -5] + 'Perhaps'`
   3. `s = s[0:6] + 'Perhaps'`
   4. `s = s - s[-5:] + 'Perhaps'`
   5. other/none/more than one

4. Strings are **immutable**, what does that mean?
5. Consider the following code:

```python
s = 'BaNana man'
s.count('an')
```

This only counts occurrences of lower case 'an'. Which of the following expressions counts the number of times 'an' occurs when case is ignored?

1. `s.count('an').lower()`
2. `s.lower().count('an')`
3. `s.count('AN'.lower())`
4. `s.lower().count('AN'.lower())`
5. other/none/more than one

6. What is the output of the following program? If there are any errors, correct them.

```python
x = 'aBrAcAdAbRa'
print(x, x.capitalize(), x.upper(), x.lower(), x.lower().count('a'))
```

7. What is the output of the following program? If there are any errors, correct them.

```python
x = 'COSC 101'
print("The 3rd character is ", x[3])
```
8. What is the output of the following program? If there are any errors, correct them.

```python
a = """apple""
b = '''banana'''
c = 2
if b[-1] == a[0]:
    print(a + b * c)
else:
    print(b + a * c)
```

9. What is the output of the following program if you enter your name and 300?

```python
name = input("Who are you? ")
total = float(input("How much money do you have? "))
x = 3
y = "cats"
print("{}, I would like to offer you the special opportunity to purchase {} {}" + 
    " at the low, low price of ${:.2f} each!".format(name, x, y, total/x))
print()
print(name + ", I would like to offer you the special opportunity to purchase", 
    x, y, "at the low, low price of $" + str(round(total/x, 2)) + " each!")
```
10. Write a program that asks the user for the name of a product, the price of a single unit, and the number of items they would like to buy. Output a receipt in the following format:

```
6 apple
@ $0.35 each
= $2.10 total
```

11. What is the output of the following program? If there are any errors, correct them.

```python
cap = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
low = 'abcdefghijklmnopqrstuvwxyz'
space = " 
```

12. Write a `print` statement to print your first and last names using the above strings (`cap`, `low`, and `space`) and string slicing.
13. What is the output of the following program? If there are any errors, correct them.

```python
a = "my name is madeline"
b = "i teach at colgate university"
print(a[:3])
print(b[19:])
print(a[8:11])
print(b[-18:-10])
```

14. Write a `print` statement that prints Madeline teaches at Colgate University using the strings defined in the previous question (a and b) with string slicing and string methods.

15. What is the output of the following program? If there are any errors, correct them.

```python
if 'a' > 'A':
    print('a is bigger than A!')
else:
    print('A is bigger than a!')
```
16. What is the output of the following program? If there are any errors, correct them.

```python
a = 'apple!
for ltr in a:
    if ltr > 'e':
        print(ltr, end=" ")
    else:
        print("x", end=" ")
```

17. Write a program that asks the user for their first and last name and prints an abbreviated name. For example:

```
What is your name? Madeline Smith
M. Smith
```

*Hint: use `find()` and slicing.*
18. What is the output of the following program? If there are any errors, correct them.

```python
a = 'hello!
for idx in range(len(a)):
    if idx % 2 == 0:
        print(a[idx], end=" ")
    else:
        print("x", end=" ")
```

19. What is the output of the following program? If there are any errors, correct them.

```python
a = 'mississippi!
accum = ''
for idx in range(len(a)):
    if a[idx] in accum:
        print(a[idx], end=" ")
    elif idx % 2 == 0:
        accum += a[idx]
print('accum = '+accum)
```

20. Assume variables \( s_1 \) and \( s_2 \) refer to strings. Write a single expression that produces the index of the second occurrence of \( s_2 \) in \( s_1 \). If \( s_2 \) does not occur twice in \( s_1 \), the expression should produce -1.

Example: if \( s_1 = "banana" \) and \( s_2 = "ana" \) the expression should return 3.
21. Write a function called `acronym` that takes a string representing a phrase or name and returns the equivalent acronym, uppercased.

```python
>>> print('computer science investigation')
'C.S.I.'
```

*Hint: use a for loop, split(), join(), and upper().*