

NAME (print): _____

Netid: _____

Community Standard Acknowledgement (signature): _____

Do NOT spend too much time on any one question.

In writing code you do not need to worry about specifying the proper `import` statements. Assume that all libraries we've discussed are imported in any code you write.

Do **not** discuss this test **with anyone** until the test is returned. Do not use the web, PyCharm, or any programming environment that checks your code.

	value	grade
Problem 1	2 pts.	
Problem 2	24 pts.	
Problem 3	20 pts.	
Problem 4	14 pts.	
TOTAL:	60 pts.	

PROBLEM 1 : (*Honor code (2 pts)*)

Print your name to acknowledge the Duke Community Standard

Print your name again to say you have read all the rules for the exam and agree to follow them (including not using the web, classmates, Pycharm, or other environments to determine answers to your program).

PROBLEM 2 : (*Short code segments (24 pts)*)

For each of the following problems, use only what is indicated to set result to a Python expression. Unless specified within the problem, DO NOT use any Python methods or simply add a string that solves the problem.

Here is an example.

Use phrase with indexing and the concatenation of two items to set result to the string 'by'.

```
phrase = 'bicycle'
result = phrase[0] + phrase[3]
```

Note this answer uses only phrase and indexing, and the concatenation of two items. It does not simply assign the string result = 'by'

PART A (3 pts)

Use phrase with indexing and concatenation to create the string 'ax'

```
phrase = 'The quick brown fox jumps over the lazy dog'
result = _____
```

PART B (3 pts)

Use lst with indexing and concatenation to create the string lie.

```
lst = ['automobile']
result = _____
```

PART C (3 pts)

Use phrase with splicing and the concatenation of two items to create the string 'hamDu'

```
phrase = 'DurhamNC'
result = _____
```

PART D (3 pts)

Use lst with indexing and the concatenation of two items to make the string 'bluecar'

```
lst = [['car', 'bike'], ['red', 'blue']]
result = _____
```

PART E (3 pts)

Use lst with indexing and the concatenation of two items to make the string 'zp'

```
lst = [['cow', 'zebra', 'lion'], 'penguin']
result = _____
```

PART F (3 pts)

Use only lst with the len function and indexing such that result is assigned the integer 9.

```
lst = [['Durham', 'Raleigh', 'Greensboro'], ['first year', 'sophomore', 'junior', 'senior']]
result = _____
```

PART G (3 pts)

Using slicing only, create a clone of lst

```
lst = [['Camille', 'Mia'], 25]
result = _____
```

PART H (3 pts)

Using only split and join functions, create the string 'b\$n\$n\$s'

```
phrase = 'bananas'
result = _____
```

PROBLEM 3 : (*Simple Functions (20 points)*)

PART A: Lab Employee Pay (10 pts)

Your lab supervisor is interested in creating a system for calculating weekly pay based on each employee's classification level and number of hours worked. Undergraduates and graduate students earn 15 per hour. Postdoctoral researchers earn 25 per hour. Staff earn 35 per hour. Undergraduates and graduate students cannot be paid for more than 20 hours in one week. Thus, if they have worked more than 20 hours in a week, then they will only be paid for the first 20 hours. Postdoctoral researchers and staff have no limit on the number of hours that they can be paid for.

Create the function `calculatePay` that has two parameters: **level** is a string representing the classification level of each lab member (**ugrad** is undergraduate, **grad** is graduate student, **post** is postdoctoral researcher, and **staff** is staff member); and **hours** is a float representing the number of hours worked. The `calculatePay` function calculates and returns the total pay, based on the information provided above.

call	returns	comment
<code>calculatePay('ugrad', 24.0)</code>	300.0	undergrads only paid for the first 20 hours at 15/hr
<code>calculatePay('staff', 35.5)</code>	1242.5	staff members paid at 35/hour with no hour limitations
<code>calculatePay('post', 10.5)</code>	262.5	postdocs paid at 25/hour

Complete the function `calculatePay` below.

```
def calculatePay(level, hours):
```

PART B: String Comparison(10 pts)

Create the function `compareString` that has two parameters: `text1` and `text2`, to determine which is greater. If `text1` is greater than or equal to `text2`, then a list named `lst` should be created that contains the two strings, with `text1` as the first item. If `text2` is greater, then `lst` should be created containing the two strings with `text2` as the first item. `compareString` should then output the list to the screen.

call	output
<code>compareString('Help', 'Halle')</code>	<code>['Help', 'Halle']</code>
<code>compareString('Help', 'HELP')</code>	<code>['Help', 'HELP']</code>
<code>compareString('HELP', 'HOLE')</code>	<code>['HOLE', 'HELP']</code>

Complete the function `compareString` below.

```
def compareString(text1, text2):
```

PROBLEM 4 : (*Reading and Debugging Programs (14 pts)*)

PART A) (8 pts)

Consider the following program:

```
1 list1 = ['red', 'black', 'green']
2 list2 = ['purple', 'gold']
3 print(list2)
4 list1[-1] = 'yellow'
5 print(list1 + list2)
6 print(list1)
7 list1.append(list2)
8 print(list1)
```

a) What is the value of list1 when the program completes?

b) What is the value of list2 when the program completes?

c) I want the result of list1 (when printed in line 8) to be ['red', 'black', 'yellow', 'gold']. What line of code would you modify?

d) Using only list1 and list2, how would you modify the line you answered in part c) such that list1 (when output in line 8) is the desired list in part c)? Note, you cannot use a string literal 'gold' in your solution.

PART B) (6 pts)

The following code checks if the last character in name2 is a substring of name1

```
1 name1 = 'North Carolina'
2 name2 = 'Sylva'
3 if name2[5] in name1:
4     print("It is there!!")
5 else:
6     print("It is NOT there!!")
```

This program doesn't execute. It produces an IndexError: string index out of range

a) What line of code contains the error?

b) What is the reason for the error?

c) What line of code fixes this error to ensure it works correctly?