

Name: _____ Andrew Id: _____

15-112 Spring 2023 Quiz 5

Up to 20 minutes. No calculators, no notes, no books, no computers. Show your work!

Do not use dictionaries, try/except, or recursion on this quiz.

1. (8 points) **Code Tracing:** Indicate what the following program prints. Place your answer (and nothing else) in the box next to the code. Make sure strings are enclosed with quotes.

```
def ct(s, t):
    r = s
    t = s.replace('a','d')
    t = t[::-1]
    print(s, t) # Do not miss this!
    r += "\n"
    for i in range(min(len(s), len(t))):
        if s[i] == t[i]:
            r += s[i] + "\n"
    return r + "bye"
print(ct('abcd', 'abd'))
```

2. (4 points) **Reasoning Over Code:** Find an argument, `s`, for the following function to cause it to return `True`. Place your answer (and nothing else) in the box below the code. Make sure strings are enclosed with quotes.

```
def roc(s):
    L = s.split('\n')
    assert(len(L) == 2)
    a = "0"
    b = ""
    for l in L:
        if len(l) != 2 or not l[0].isdigit() or not l[1] in 'hi':
            return False
        a += l[0]
        b += l[1]
    return a + b == "042ih"
```

3. (8 points) **Free Response:**

Write the function `isValidCMUEmail(s)` that takes in a lowercase string `s` and returns `True` if `s` is a valid CMU e-mail address and `False` if it is not.

A valid CMU e-mail address is of the form `<account name>@<domain name>` and satisfies the following constraints:

- The domain name ends with `cmu.edu`
- The domain name can have more sub-domains separated by dots. For example, `cs.cmu.edu`, `it.qatar.cmu.edu`
- Each additional sub-domain (if present) is non-empty and contains only alphabetical letters.
- The part that precedes the `@` (the account name) is non-empty and contains at least one alphanumeric character and zero or more valid symbols (dashes `-`, underscores `_`, or dots `.`)

For example:

```
# valid CMU addresses
assert(isValidCMUEmail("example@andrew.cmu.edu") == True)
assert(isValidCMUEmail("15112@cs.qatar.cmu.edu") == True)
assert(isValidCMUEmail("15.112-s23_staff@cmu.edu") == True)
assert(isValidCMUEmail("thedean@cmu.edu") == True)
# NOT valid CMU addresses
assert(isValidCMUEmail("what's_this?@cmu.edu") == False)
assert(isValidCMUEmail("blabla@notvalid.edu") == False)
assert(isValidCMUEmail("inv@lid@ddress") == False)
assert(isValidCMUEmail("notgood@gmail.com") == False)
assert(isValidCMUEmail("notok@..cmu.edu") == False)
```

Hint: The following built-in `string` methods may be useful: `isalpha()` that checks if the characters are alphabetical letters, `isalnum()` that checks if the characters are alphanumeric, and `split`.

You may continue your work on the back of this page...

**THIS PAGE IS INTENTIONALLY LEFT BLANK.
NOTHING ON THIS PAGE WILL BE GRADED.**