

CSc 110

Final Exam Review and Wrap-Up

Adriana Picoral (she/her/hers)

Announcements

- Final exam: Friday (December 9) at **1pm** (2 hours)
- Review session: 5pm on Thursday December 8th
 - Physics-Atmospheric Sciences (PAS) 201

Write a function

Write a function named **even_sum**

This function should take one parameter

- A 2D list of numbers

The function should sum the even numbers within the 2D list, and return the sum.

```
even_sum([[2, 3, 1, 2], [2, 4, 5, 2, 3], [3, 2, 6, 3]]) # 20
```

Write a function

Write a function named **even_average**

This function should take one parameter

- A 2D list of numbers

The function should return the average of the even numbers within the 2D list

```
even_sum([[2, 3, 1, 2], [2, 4, 5, 2, 3], [3, 2, 6, 3]]) # 2.857142857142857
```

Write a function that does the following:

1. Its name is `print_all`.
2. It takes two integer arguments, `x`, and `y`.
3. It prints all the integer values between `x` and `y`, inclusive.
4. It should print those numbers in ascending order.
5. Note that `x` may, or may not, be less than `y`.
6. Do not use a `for` loop.

```
def print_all(x, y):
```

Write a function that does the following:

1. Its name is `keep_vowels`.
2. It takes a single string as its parameter.
3. It returns the string that represents the parameter with all the non-vowels removed.
4. Do not use `replace` or any other built-in string manipulation routines.

```
keep_vowels("hello") # "eo"  
keep_vowels("aeiou") # "aeiou"  
keep_vowels("") # ""
```

```
def keep_vowels(s):
```

Write a function that does the following:

1. Its name is `collect`.
2. Its parameter is a list of values.
3. It will create and return a dictionary.
4. The return dictionary will have the values in the original list as keys.
5. The values associated with each key will be a list of the indices in the original list where that key appeared.

```
collect([]) # {}
```

```
collect([1, 2, 3, 1, 2]) #{1: [0, 3], 2: [1, 4], 3: [2]}
```

```
def collect(numbers):
```

Helping with Courses

- Are you interesting in helping with 101, 110, or 120 in the future?
- After or during 120, apply to help with the intro courses!

Helping with Courses

- Are you interesting in helping with 101, 110, or 120 in the future?
- After or during 120, apply to help with the intro courses!
- On that note . . .

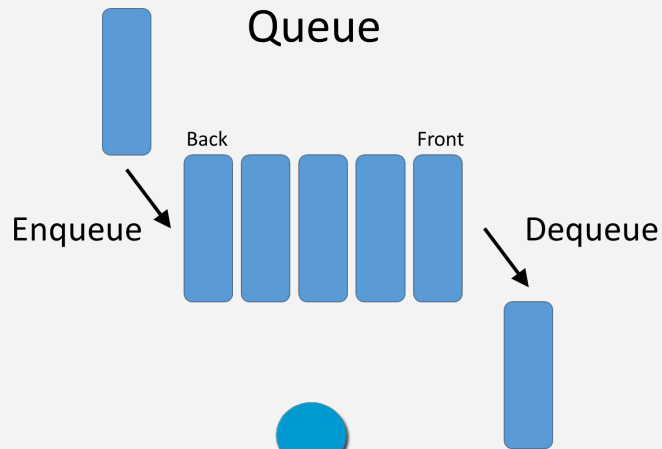
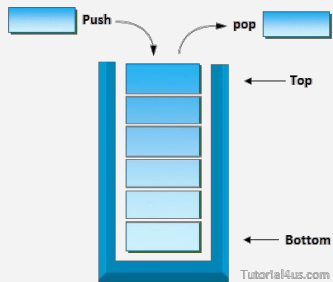
Thanks TAs!

Why keep going with CS?

- Great job opportunities as software engineer/developer
- Also is a GREAT major or minor to pair with many other degrees
 - Business
 - Natural Sciences (Bio, Chem, Geos, etc)
 - Engineering
 - Medicine
 - Others!

CS 120

- Python - based
- Programming, programming, programming!
- Topics include:
 - Data structures: Lists (Arrays), Stacks, Queues, Trees, Linked-Lists
 - Classes and Objects
 - Invariants
 - Testing



Computer Science

- Algorithms
- Security
- Data Vis
- Parallel + Distributed Computing
- Compilers
- Databases

Thank You

- Thanks to you, the students!
- Feel free to reach out in the future