

# CIS 122 Fall 2015 Project 4

## Due Monday, October 26, 11:59 PM

### Briefly

Submit four Python 3 programs  
Your programs are worth a total of 25 points

Test your programs -- did they work right? -- before uploading to Canvas.

Your programs should contain code that tests your functions.

### Project4-get-numbers.py 5 points:

#### Learning objective

**Short simple single purpose functions** simplify your programs and reduce errors.

Define a **get\_int(prompt, minval, maxval)** function that uses the prompt with the input function and returns an integer value.

If the number typed in is smaller than minval or bigger than maxval, print an error message and try again.

Your code must **test your get\_int function**:

For example

```
for i in range(5):
    prompt = "Type a number between 1 and 100: "
    number = get_int(prompt, 1, 100)
    print("You entered", number)
#end for
```

Try values like 0, 59, 99, 100, 1

#### Rubric

- 1 point Include a docstring as first line(s) of your def.
- 1 point When number typed is too small print an error message.
- 1 point When number typed is too big print an error message.
- 2 points get\_int must return an int (whole number), not a string.

### Project4-guess.py 7 points:

#### Learning objectives

**While loop** works with user interactions.  
Learn basic use of **random numbers**.

Again, use **get\_int(prompt, minval, maxval)** function (copy and paste can help).

This fragment of code will help:

```
import random
answer = random.randint(1, 100)
# randomly chooses a number from 1 through 100.
```

Ask your user to guess a number between 1 and 100.

If correct, print a message of congratulations and exit your while loop.

if the guess is too large, print something like "That is too big; try a smaller number"

if the guess is too small, print something like "That is too small, try a larger number"

#### Rubric

- 1 point Correctly assign a random number to answer.
- 1 point Uses get\_int(prompt, minval, maxval) to get user's guess.
- 1 point When number typed is too small print a message.
- 1 point When number typed is too big print a message.
- 1 point When number typed matches the answer, print a message of congratulations and exit your while loop.
- 2 points while loop correctly works with user's guesses

### Project4-miles-list.py 8 points

#### Learning objectives

**Create a list, starting from an empty list. Check your get\_float function for getting decimal numbers.**

Define a **get\_float(prompt, minval, maxval)** function that uses the prompt with the input function and returns a floating point number. The numbers are number of miles you rode your bike.

If the number typed in is smaller than minval or bigger than maxval, print an error message and try again.

After you gather 5 or 6 numbers and put them in a **miles** list, **print the sum of the miles entered**. Hint: the sum function is useful.

#### Rubric

- 1 point Include a docstring as first line(s) of your def.
- 1 point get\_float correctly checks that numbers entered are within minval and maxval.
- 2 points get\_float returns a float (decimal number), not a string.
- 2 points Create a miles list from the floating point numbers.
- 2 points Get the correct total of all the miles in the miles list.

### Project4-buildings.py 5 points

#### Learning objective

**Create a buildings list**, starting from an empty list.

You could use the names of buildings at UO, or Taj Mahal, Louvre, White House, etc

Use a while loop to gather the building names.

After gathering the building names, print each building on a separate line.

#### Rubric

- 1 point Start from an empty list
- 3 points while loop gathers names using input function
- 1 point for loop prints each building name, one name per line.

## Bonus Challenge

### Project4-challenge.py

Revise the guess program to keep track of all the guesses entered.

Do not let a user guess a previous wrong number again; instead note that number was already used; please try a new number.

3 points