Review of Loop Practice

Write a *while loop* that will compute the sum of the first $n$ positive odd numbers. For example, if $n$ is 5, you should compute $1 + 3 + 5 + 7 + 9$.

While Loop General Algorithm

```
while some condition is true :
    # Do this code
    # Something here should modify the condition above
```

```c
# AIM Machine - keeps asking for your PIN until you enter the right one
pin = int(input("What is your PIN? "))
while pin != 1234:
    print("Wrong PIN")
    pin = int(input("What is your PIN? "))
```
Loops that Count

These loops are equivalent.

```python
cnt = 1
while cnt <= 100:
    if cnt % 2 == 0:
        print(cnt, "is even")
    else:
        print(cnt, "is odd")
    cnt += 1
```

```python
cnt = 1
while cnt <= 100:
    print(cnt, "is odd")
    print(cnt + 1, "is even")
    cnt += 2
```

Practice

Design an algorithm for finding the greatest common divisor of 2 integers.

The GCD of 16 and 24 is 8.
The GCD of 4 and 2 is 2.

- Write your solution in pseudocode.
- Work in groups of 2-3 people.
*Hint – your algorithm should include a while loop.

Standard Library Functions and the import Statement

- Just like we have built-in functions like print and input, we also have libraries of other pre-written functions.
- Modules: files that stores functions of the standard library
  - Help organize library functions not built into the interpreter
  - Copied to computer when you install Python
- To call a function stored in a module, need to write an import statement
  - Written at the top of the program
  - Format: `import module_name`

The math Module

- `math` module: part of standard library that contains functions that are useful for performing mathematical calculations
  - Typically accept one or more values as arguments, perform mathematical operation, and return the result
  - Use of module requires an `import math` statement
The `math` Module

- The `math` module defines variables π and e, which are assigned the mathematical values for π and e
  - Can be used in equations that require these values, to get more accurate results
- Dot notation: notation for calling a function or variable belonging to a module
  - Format: `module_name.function_name()`
  - Format: `module_name.variable_name`
- Variables must also be called using the dot notation
  - Example: `circle_area = math.pi * radius**2`

Generating Random Numbers

- Random number are useful in a lot of programming tasks
- `random` module: includes library functions for working with random numbers
- `randint` function: generates a random number in the range provided by the arguments

Generating Random Numbers (cont’d.)

- The `random` function returns a value
- A random number in the range of 1 through 10 will be assigned to the number variable
- A random number in the range of 1 through 10 will be displayed.
Using Random Numbers Example

# This program displays a random number
# in the range of 1 through 10.
import random

def main():
    # Get a random number.
    number = random.randint(1, 10)
    # Display the number.
    print('The number is:', number)

    # Call the main function.
    main()