Python Cheat Sheet

This document contains a list of the tools that will be required to write a script for each problem in the algorithm project. The goal of the project is to develop general solutions for mathematics problems and convert the general solutions to automated programs. A more comprehensive set of detailed instructions can be found at: https://www.w3schools.com/python/default.asp

Printing Text

In Python, text is written using the `print()` function. For example, if I wanted to print "Hello World":

```
print("Hello World")
```

For the programs we write in class, we will start by printing a description of what the program does.

Comments

Comments are used in Python for the script writer to note what they are doing and why they are doing it. When executing a program, comments are "invisible." The # sign makes a comment. For example:

```
print("Hello World") # This will print "Hello World"
```

Assigning Variables

In Python, you can assign variables using the = operator. For example, say I want to assign a height of 5 to the variable `h`:

```
h = 5 # This assigns 5 to the variable h
```

Data Types

We will deal with four different data types in Python:

- **string:** text data
- **int:** integer
- **float:** real number
- **bool:** Boolean (logic values: True/False)

Math Operations in Python

In Python, math operations are very similar to those on your calculator. Below is a table summarizing basic operations with illustrated examples:

Operation	Mathematics	Python
Addition	$a + b$	a+b
Subtraction	$a - b$	a-b
Multiplication	$a \times b$	a*b
Division	$\frac{a}{b}$	a/b
Exponentiation	a^b	a**b

We will also use the **modulus** operator, %. This operator finds the remainder of a quotient. For example:

```
5%4 # This will return 1, as 5/4 has 1 remainder
5%5 # This will return 0, as 5/5 has 0 remainder
```

The `math` package will also be required. The `math` package is a pre-written set of functions for Python users that helps with math operations. To use the `math` package:

```
import math # imports the math package
pi = math.pi # Create pi variable
```

Taking Input from the User

In order to take input from a user, we use the `input()` function. We also want to define the type of data the user is inputting. Text can be added directly in the input function. For example, suppose I want the user to input their age:

```
age = int(input("Input your age: "))
```

This line will assign what the user inputs to the variable `age` and convert the data type into an integer.

If Statements

If statements are used when testing some sort of condition. `if` takes in some sort of operation and evaluates as True or False. "Else" is what executes if the original statement is false. For example, suppose I wanted to print "above five" if a variable x is above five:

```
x = 7
if x > 5:
    print("above five")
else:
    print("not above five")
```

In the terminal, "above five" will print.

For/While Loops

Loops are used when doing iterative operations. While loops continue to execute until a condition is met. For example, say I wanted to print the numbers up to 50 quickly:

```
count = 1 # counter variable
while count < 51:
    print(count) # prints number
    count = count + 1 # adds 1 to the counter
```

In the terminal, this will print each number from 1 to 50. For loops work very similar to while loops. They run a statement over a defined number of iterations. For example, say I wanted to print the numbers up to 50 quickly:

```
for i in range(1, 50): # i=1, then i=2,...
    print(i)
```