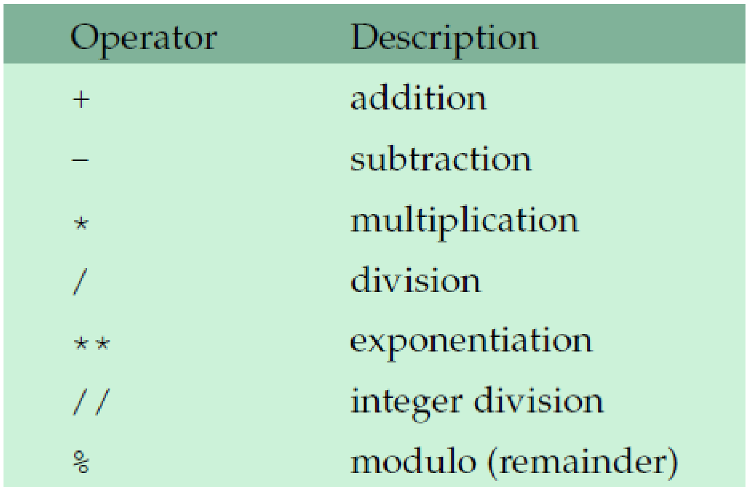
College of Engineering & Technology
Computer Techniques Engineering Department
Artificial Intelligence – Stage 3
(Python)

Number (Integers and Decimal), Math Operators, Order of Operations, and Math Functions

1. **Numbers (Integers and Decimal):** because of the way computer chips are designed, integers and decimal numbers are represented differently on computer. Decimal numbers are represented by what are called floating point numbers. The important thing to remember about them is you typically only get about 15 or so digits of precision. It would be nice if there were no limit to precision, but calculation run a lot more quickly if you cut off the numbers at some point. On the other hand, integers in Python have not restrictions. They can be arbitrarily large.

   For decimal numbers, the last digit is sometimes slightly off due to the fact that computers work in binary (base 2) whereas our human number system is based 10. As an example, mathematically, we know that the decimal expansion of 7/3 is 2.333……, with the threes repeating forever. But when we type 7/3 into the python we get. 2.3333333333333335. This is called *roundoff error*.

2. **Math operators:** here is a list of the common operators in pythons:



Operator	Description
+	addition
–	subtraction
*	multiplication
/	division
**	exponentiation
//	integer division
%	modulo (remainder)

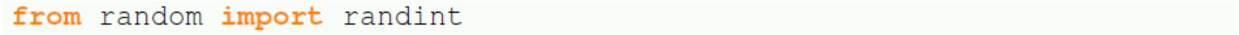
- **Exponentiation:** python uses **
- **Integer division (//):** for positive integer it throws away the decimal part of the result, for instant, while 8/5 is 1.6, we have 8//6 equal to 1.

- **Modulo:** the modulo operator, %, returns the remainder for a division.
- **The rest of the operators (+, - , * , /) are standard.**

3. **Order of operations:** Exponentiation gets done first, followed by multiplication and division (including // and %), and addition and subtraction come last. This comes into play in calculating an average. Say you have three variables x, y , and z, and you want to calculate the average of their values. The expression x+y+z/3 would not work. Because division comes before addition, you would actually be calculating $(x+y+\frac{z}{3})$ instead of $(\frac{x+y+z}{3})$. This is easily fixed by using parentheses: $(x+y+z)/3$.

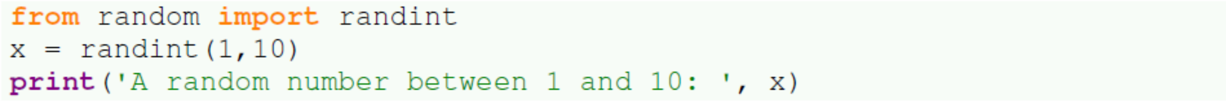
4. **Random numbers:** python comes with a module, called *random*, that allows us to use random numbers in our programs. Before we get to random numbers, we should first explain what a *module* is. The core part of the Python language consists of things like for loops, if statements, math operators, and some functions, like print and input, everything else is contained in modules, and if we want to use something from a module, we have to first *import* it. This is telling Python that we want to use it.

   At this point, there is only one function, called *randint*, that we will need from the random module. To load this function, we use the following statement:



```
from random import randint
```

Using randint is simple, randint(a,b), will return a random integer between a and b including both a and b. Here is a short example of randint:

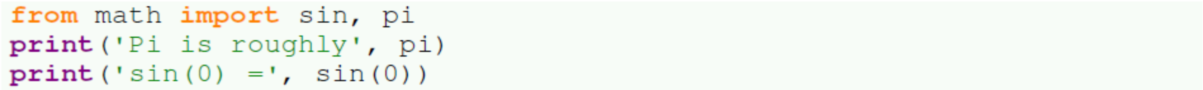


```
from random import randint
x = randint(1,10)
print('A random number between 1 and 10: ', x)
```

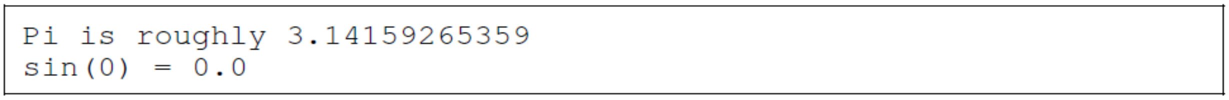
```
A random number between 1 and 10: 7
```

The random number will be different every time we run the program.

5. **Math Functions:** Python has a module called math that contains familiar math functions, including: sin, cos, tan, exp, log, log10, factorial, sqrt, floor, and ceil. There are also the inverse trig functions trig functions, hyperbolic functions, and the constants pi and e. Here is a short example:

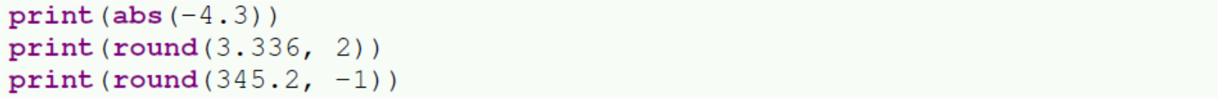


```
from math import sin, pi
print('Pi is roughly', pi)
print('sin(0) =', sin(0))
```

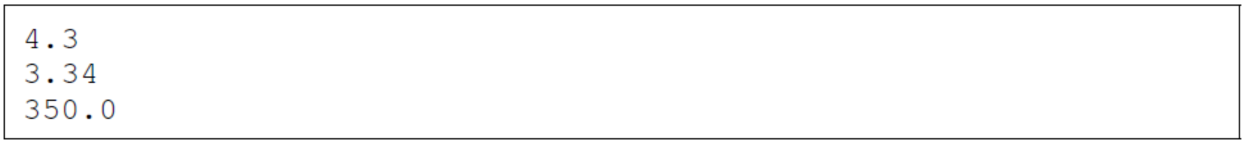


```
Pi is roughly 3.14159265359
sin(0) = 0.0
```

6. **Build-in math function:** there are two built in math functions, abs (absolute value) and round that are available without importing the math module. Here are some examples:



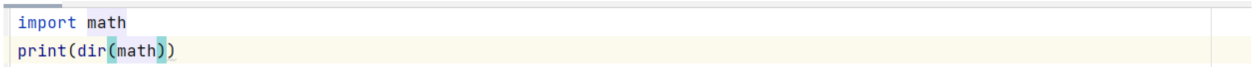
```
print(abs(-4.3))
print(round(3.336, 2))
print(round(345.2, -1))
```



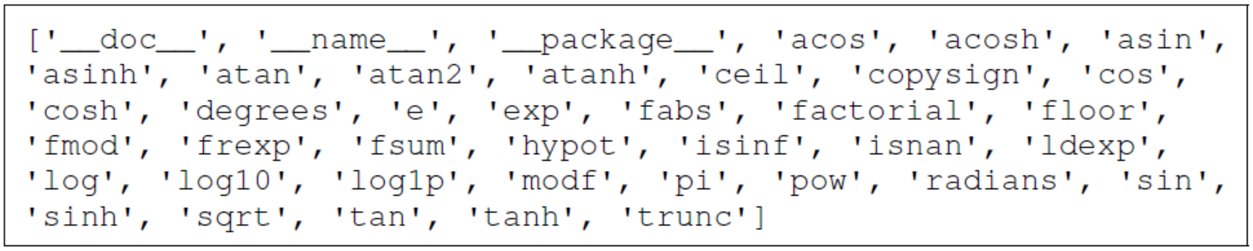
```
4.3
3.34
350.0
```

The round function take two arguments, the fist the number to be rounded and the second is the number of decimal places to round to. The second argument can be negative.

7. **Getting help for Python:** there is documentation built into Python. To get help on the math module, for example, go to the Python shell and typey the following lines:

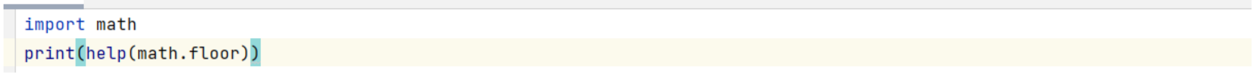


```
import math
print(dir(math))
```



```
['__doc__', '__name__', '__package__', 'acos', 'acosh', 'asin',
'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'copysign', 'cos',
'cosh', 'degrees', 'e', 'exp', 'fabs', 'factorial', 'floor',
'fmod', 'frexp', 'fsum', 'hypot', 'isinf', 'isnan', 'ldexp',
'log', 'log10', 'log1p', 'modf', 'pi', 'pow', 'radians', 'sin',
'sinh', 'sqrt', 'tan', 'tanh', 'trunc']
```

This gives a list of all the functions and variables in the math module. You can ignore all of the ones that start with underscores. To get help on a specific function, say the floor function, you can help(math.floor). Typing help(math) will give you help for every thing in the math module.



```
import math
print(help(math.floor))
```

EXERCISES

1. **Write a program that generates and print random integers between 3 and 6.**
2. **Write a program that generates a random number, x, between 1 and 50, a random number y between 2 and 5, and computes x^y.**
3. **Write a program that generates a random decimal number between 1 and 10 with two decimal places of accuracy. Examples are 1.23, 3.45,9.80 and 5.00.**
4. **Write a program that asks the user to enter two numbers, x and y, and computes $\frac{|x-y|}{x+y}$**
5. **Write a program that asks the user for a number of second and prints out how many minutes and second that is. For instance, 200 seconds is 3 minutes and 20 seconds. Hint use the // operator to get minutes and the % operator to get seconds.**
6. **Write a program that asks the user to enter a weight in kilograms. The program should convert it to pounds, printing the answer rounded to the nearest tenth of a pound.**
7. **Write a program that asks the user for number and then prints out the sine, cosine, and tangent of that number.**
8. **Write a program that asks the user to enter an angle in degrees and prints the sine of that angle.**