FIT1045/53 Introduction to Programming and Python – Workshop 2.

Objectives

The **objectives of this workshop** are:

- To get familiar with Python.
- To be able to perform iteration and selection in Python.
- To be able to perform basic string and numerical manipulation.
- To be able to import and use the math package and random package.
- To be able to do simple input/output.

Useful Material

Selection and Iteration: `https://docs.Python.org/3/tutorial/controlflow.html`

Strings: Some introductory information is available at `https://docs.Python.org/3/tutorial/introduction.html#strings` and more at `https://docs.Python.org/3/library/stdtypes.html#string-methods`.

Preparation

If you have not completed the coin flipping exercise from Workshop 1, make sure you complete it before attending this weeks workshop.

Flipping Coins Continued...

In this task we will be continuing in our use of the random library from Workshop 1. This task will be divided up into 3 parts. It is important that you complete this task before continuing with the rest of the workshop.

Task 1(a)

Write a program that takes a number n as input from the user and simulates n coin flips printing the results each time. Using your knowledge of selection, improve upon the output suggested in Workshop 1, Task 2(b).

Hint: You will need to ensure that you convert the string input into numbers

Example:

```
How times would you like to flip the coin? 5
The coin came up heads.
The coin came up tails.
The coin came up tails.
The coin came up heads.
The coin came up heads.
```

Task 1(b)

Modify the program from task 1(a) to store the number of heads and tails in variables. Once you have generated the variables, print the total number of heads and the total number of tails. Calculate the ratio of heads to the total coin flips.[1] Is this ratio what you would expect, what happens to this ratio as you change the probability of the coin coming up heads? What happens as the number of coins being flipped becomes large?

[1] The ratio of two numbers, A and B, is given by $\frac{A}{B}$

Task 1(c)

Up to this point you have been using the ‘random.random()’ function to generate a random number between 0 and 1. Using the documentation from the random library, use ‘random.randrange(a)’ to extend your program from part 2 so that it uses 3 sided coins.[2]

Note: What happens if you call ‘random.randrange()’? Discuss these results with the person sitting next to you and don’t forget to ask for help if you get stuck.[3]

Operations on sequences of integers

In mathematics, summation is where a group of numbers are added together to form a single numeric result and is represented using the greek symbol sigma Σ. We can use this to compactly represent a summation for a large group of numbers, for example $\sum_{i=1}^{3} i$

> *Where i is the subject and starts at 1, and the sequence ends at 3. Sigma then denotes the sum of the sequence.*

Beneath the sigma we see $i = 1$ which suggests we are using i as an index and it begins at 1; above the sigma we see the last value that i will take. Based on this:
$\sum_{i=1}^{3} i = 1 + 2 + 3 = 6$ and $\sum_{i=0}^{5} i = 0 + 1 + 2 + 3 + 4 + 5 = 15$
You are not limited to single variables in summations, we could equally write something like this:
$\sum_{i=1}^{3} 2i^3$ which is $2 \times 1^3 + 2 \times 2^3 + 2 \times 3^3 = 2 + 2 \times 8 + 2 \times 27 = 72$

Task 2(a)

Prepare a python program which can compute the solution for the following series (using loops): $\sum_{i=1}^{10} 3i$
Hint: You can check your answer with excel or another spreadsheeting system

Task 2(b)

Let’s make this a little more interesting, let’s allow the start and end values to be chosen by the user. For example:

```
Where should i start? 3
Where should i stop? 5
The result for sum of 3i from 3 to 5 is: 36
```

This is as $\sum_{3}^{5} 3i = 3 \times 3 + 3 \times 4 + 3 \times 5 = 9 + 12 + 15 = 36$

[2] https://docs.python.org/3/library/random.html

[3] By this we mean call ‘random.randrange()’ without a parameter.

Task 2(c)

Let's bring in some selection as well. For this task, let's try a different summation $\sum_{i}^{n} 2i^2 + 4i$ and continue asking for a start and stop value but we will change something else. This time, only add items where the i value is even [4].
For example $\sum_{i=1}^{n=5} 2i^2 + 4i = 170$ however after excluding odd numbers the result is 64 which comes from $0 + (2 \times 2^2 + 4 \times 2) + 0 + (2 \times 4^2 + 4 \times 4) + 0 = (2 \times 4 + 8) + (2 \times 16 + 16) = 16 + 48 = 64$ as 1, 3 and 5 were all odd numbers.
Extend your program so the user can suggest what valid i values should be divisible by (e.g. by 3 instead of 2). For instance:

```
Where should i start? 3
Where should i stop? 5
Valid i values are those divisible by... 3
The result for sum of 2i^2+4i from 3 to 5 (with i values divisible by 3) is: 30
```

This is as only 3 is divisible by 3 in the range of 3 to 5 inclusive. So the result is $2 \times 3^2 + 4 \times 3 = 2 \times 9 + 12 = 30$

Task 2(d)

Let's try something a little cooky now, let's have a sum of sums. Prepare a python program which will compute the value of the following summation: $\sum_{i=1}^{n}(\sum_{j=1}^{i}(2i^2 + 4j))$. For example, if $i = 1$ and $n = 3$ the result is
$\sum_{i=1}^{3}(\sum_{j=1}^{i}(2i^2 + 4j))$
$= \sum_{j=1}^{1}(2 \times 1^2 + 4j) + \sum_{j=1}^{2}(2 \times 2^2 + 4j) + \sum_{j=1}^{3}(2 \times 3^2 + 4j)$
$= (2 \times 1 + 4 \times 1) + \Big((2 \times 2^2 + 4 \times 1) + (2 \times 2^2 + 4 \times 2)\Big) + \Big((2 \times 3^2 + 4 \times 1) + (2 \times 3^2 + 4 \times 2) + (2 \times 3^2 + 4 \times 3)\Big)$
$= (2 + 4) + \Big((2 \times 4 + 4) + (2 \times 4 + 8)\Big) + \Big((2 \times 9 + 4) + (2 \times 9 + 8) + (2 \times 9 + 12)\Big)$
$= 6 + (12 + 16) + (22 + 26 + 30)$
$= 6 + (30) + (78)$
$= 112$
For instance:

```
Where should i start? 1
Where should i stop? 3
The result for sum of the sums of 2i^2+4j from 1 to 3 is: 112
```

Task 2(e)—Optional (attempt Task 3 first if you are running short on time)

Extend your program so that the start and stop values are random and that the user must give the correct result which matches this. The user should keep guessing until they give up (by entering "quit") or get it correct.

```
You will be guessing the sum of the sums of 2i^2+4j from i=2 to 8 and j=1 to i.
What is the result? quit
[program ends]

You will be guessing the sum of the sums of 2i^2+4j from i=1 to 3 and j=1 to i.
What is the result? 100
Incorrect
What is the result? 150
Incorrect
What is the result? 112
Correct
[program ends]
```

Task 3: Rock, Paper, Scissors (lizard, spock)

Rock, paper, scissors is a game in which who people battle by choosing either rock, paper or scissors. Paper beats rock, rock beats scissors and scissors beats paper.

1. Write a function `beats(choice1, choice2)` Which returns `True` if choice1 beats choice2 and `False` otherwise. `beats(‘‘rock’’, ‘‘scissors’’)` would return `True`

[4] you can tell if a number if even if the remainder after dividing by 2 is 0 (eg. num%2==0)

2. Write a program that randomly chooses either rock, paper or scissors and compares this to the user input. The program should then announce the winner or a draw if there is no winner. The game should keep going until the user enters ‘quit’. Use your function from part 1 in this program.

Example:

```
What do you choose? Rock
Your opponent chose paper... You lose!
```