Worksheet 1.1: Coin Tossing

An Example Assignment Attempt by Eric Johnson

In this document, we'll provide an example of an assignment attempt based on part of the first worksheet on coin flipping. It's worth noting that for attempts, you can also submit a Jupyter notebook (although a PDF is preferred), the only consideration then is that you should be **extra clear** about what problem you are attempting.

As a reminder, for assignment attempts, we just need to provide evidence that we've specifically thought about every (sub)problem in the assignment. In this sample attempt, we'll provide different examples of excellent, sufficient, and insufficient attempts. You only need to achieve **sufficiency** - excellence is neat, but not the point of these exercises.

1. (a) **Use the `help()` function to read about the various arguments and keyword arguments that `np.random.rand` takes. Describe how to use this function in your own words.**

Copying over the question is recommended!

Insufficient Response

I don't know. I'll Google later.

This is insufficient because it is vague and doesn't indicate what specifically the student doesn't know or what they are Googling!

Sufficient Attempt

I typed `help(np.random.rand)` but I found the documentation hard to understand. I don't know how it works.

This is sufficient because it indicates specifically what the student is having trouble with: reading documentation.

Excellent Attempt

I typed `help(np.random.rand)` but I found the documentation hard to understand. I Googled the function, and now I see that the function generates a numpy array of uniform random numbers between 0 and 1 that has a specified shape.

This is excellent because it provides a completed solution - a description of the function - and it indicates how the student worked through the problem.

(b) **Generate many random numbers (how many is “many”?) and print their maximum and minimum to the screen. How many of your numbers are bigger than 0.5? How many are bigger than 0.6? 0.95?**

Insufficient Response

```
print(max(x))
print(min(x))
```

This is *just barely* insufficient because we don’t know what `x` is or why the student is printing max/min of it.

Sufficient Attempt

```
rands = np.random.rand(100)
print(max(rands))
print(min(rands))
```

This is sufficient because we can clearly see that the student has printed the max and min of the random numbers.

Excellent Attempt

I think I need to ask about how to determine how many numbers are bigger than 0.5/0.6/0.95 - something to do with conditional statements? I did get the random numbers and print their max and min though:

```
rands = np.random.rand(100)
print(rands)
print(max(rands))
print(min(rands))
```

This is excellent because it explains what the student did and proposes a question to help them answer the parts they cannot complete.

(c) **Based on your generated numbers, what is the likelihood of getting a number bigger than 0.75?**

Insufficient Response

```
x > 0.75
```

This is insufficient because we don't know what x is or why the student is comparing it.

Sufficient Attempt

```
rands = np.random.rand(100)
print(np.sum(rands > 0.75))
```

This is sufficient because we can see that the student almost has a solution.

Excellent Attempt

I am stuck on how to compare the numbers to show this in Python, but the probability of being bigger than 0.75 should be 0.25 for uniform random numbers

This is excellent even though the student couldn't complete the problem or write any code, they identified what they didn't know (conditional statements in Python) and made a comment on the answer using what they did know (uniform random numbers).

(d) **To what number would I want to compare my random numbers so that 60% of the time I get a `True` and the rest of the time I get a `False`?**

Insufficient Response

0.3

This is insufficient because it's the wrong answer and we have no idea why the student thinks this is an answer.

Sufficient Attempt

0.6

This is (barely) sufficient because it's technically the correct answer, although there's no explanation.

Excellent Attempt

I think you would want to compare to 0.6, because like in the last problem, 60% of uniform random numbers should be smaller than 0.6 and 40% should be bigger.

This is excellent because it provides a full, explained solution.

(e) **Explain in words how you can use `np.random.rand` to simulate a coin where $P(Heads) = 0.6$.**

Insufficient Response

blank

This is insufficient because nothing has been written.

Sufficient Attempt

```
    rands = np.random.rand(100)
    coins = rands < 0.6
```

This is sufficient because we can see that the student has provided code that simulates a coin, even though they haven't provided a correct solution that describes this code in words.

Excellent Attempt

Based on the previous problems, I think you can use `np.random.rand` to generate uniform random numbers between 0 and 1 and then somehow compare them to 0.6 so that the ones smaller than 0.6 are True and the ones bigger are False. We can then make a set of "Heads" and "Tails" by making a new list with a "Heads" every time we see True and a "Tails" when we see False, as we saw in the Tutorial practice problems.

This is excellent even though the student couldn't write any code, they provided a complete solution to the problem.