1. Which of the following snippets of Python code are equivalent the following R code

```
draws <- sample(c(0, 1, 2), steps, replace=TRUE)
```

where steps is an integer $\geq 0$. All R vectors should be translated to Python lists.

A
```
draws = []
for i in range(steps):
draws += randint(0, 2)
```

B
```
draws = []
for i in range(steps):
draws.append(randint(0, 2))
```

C
```
draws = step * randint(0, 2)
```

D
```
draws = []
for i in range(steps):
draws.append(randint(0, 3))
```

2. Which of the following snippets of Python code are equivalent the following R code

```
draws <- sample(c(0, 1, 2), steps, replace=TRUE)
```

where steps is an integer $\geq 0$. All R vectors should be translated to Python lists.

A
```
draws = []
for i in range(steps):
draws += randint(0, 2)
```

B
```
draws = []
for i in range(steps):
draws.append(randint(0, 2))
```

C
```
draws = step * randint(0, 2)
```

D
```
draws = []
for i in range(steps):
draws.append(randint(0, 3))
```

Answer: B
This code generates a list of length steps of random numbers drawn from 0, 1, 2. Those random numbers are equivalent to those generated by `randint(0, 2)`. We translate this vectorized code into an equivalent loop that generates a list of random numbers. Note that answer A would have an error because we can't concatenate an integer and a list.
3. Which of the following snippets of Python code are equivalent to the following R code

\[
\text{fixed} \leftarrow \frac{\text{counts}[1]}{1: \text{length}(\text{counts})}
\]

where counts is a non-empty vector of numbers. All R vectors should be translated to Python lists.

A) \[ \text{fixed} = \frac{\text{counts}[1]}{(\text{len}(\text{counts}) - 1)} \]
B) \[ \text{fixed} = \frac{\text{counts}[0]}{\text{len}(\text{counts})} \]
C) \[ \text{fixed} = [] \\
\text{for } i \text{ in range(len(counts))}: \\
\text{fixed.append(counts}[1]/(i+1)) \]
D) \[ \text{fixed} = [] \\
\text{for } i \text{ in range(len(counts))}: \\
\text{fixed.append(counts}[0]/(i+1)) \]

4. Which of the following snippets of Python code are equivalent to the following R code

\[
\text{fixed} \leftarrow \frac{\text{counts}[1]}{1: \text{length}(\text{counts})}
\]

where counts is a non-empty vector of numbers. All R vectors should be translated to Python lists.

A) \[ \text{fixed} = \frac{\text{counts}[1]}{(\text{len}(\text{counts}) - 1)} \]
B) \[ \text{fixed} = \frac{\text{counts}[0]}{\text{len}(\text{counts})} \]
C) \[ \text{fixed} = [] \\
\text{for } i \text{ in range(len(counts))}: \\
\text{fixed.append(counts}[1]/(i+1)) \]
D) \[ \text{fixed} = [] \\
\text{for } i \text{ in range(len(counts))}: \\
\text{fixed.append(counts}[0]/(i+1)) \]

Answer: D

R is 1-indexed so that counts[1] is the first element of counts. The sequence 1:length(counts) is 1, 2, ..., length(counts), i.e. inclusive end, or range(1, len(counts)+1). The result is a vector so we need to perform the computation as a loop.