

Pardis Sadatian Moghaddam

Panther ID: 002722641

HW1: Estimate the expected number of times you need to toss an unbiased coin in order to observe heads in a row using either a formula or a Monte-Carlo algorithm - the formula or algorithm should be described.

We toss an unbiased coin, and the tail appears. So, the head does not appear, and we need to do x more to reach the head.

Number of flips = $(x + 1)$

Probability = $(1/2)$

If the head happened for the first flip and the tail the next time, we need to do x more to reach what we want. We already wasted two flips, and we need to do more. That is the reason we consider it $x + 2$.

Number of flips = $(x + 2)$

Probability = $(1/4)$

And the best case scenario is we get two heads in the first two flips.

Number of flips = 2

Probability = $(1/4)$

$$X = (1/2)*(X+1) + (1/4)*(X+2) + (1/4)*2 = 6$$

main.py



Run

```
1 import random
2
3 number = list()
4
5 for test in range(1000):
6     #0 for tail and 1 for head
7     one = random.randint(0,1)
8     i = 2
9     while (True):
10         #0 for tail and 1 for head
11         two = random.randint(0,1)
12         if one and two:
13             number.append(i)
14             break
15         i+=1
16
17 result = sum(number) / (len(number))
18 print(result)
```

6.182

> |