University of Southern California

EE511

Simulation Methods for Stochastic Systems

Project #1

Coin Flips

BY

Akshay Deepak Hegde

USC ID: 8099460970

[hegdeaks@usc.edu](mailto:hegdeaks@usc.edu)

Introductory Theory:

Bernoulli trials:

Bernoulli trial is a trial where only two outcomes are possible. Trials like flipping a coin can be called Bernoulli trial. If you flip a coin, only two outcomes are possible, that is, head and tail. Hence, flipping of coin is a Bernoulli trial.

Suppose each of our trials has a probability p of success,

Then the probability of getting exactly k successes out of n trials is = C(n, k) \* p^k \* (1-p)^(n-k)

If p is the probability of success, then the probability of failure, q=1−p.

The characteristics of a Bernoulli trial are:  
  
1) It can have only two outcomes, that can be labelled as success and failure.  
  
2) Probability of success and failure remains same through each trial.  
  
3) The trials are independent of each other.  
  
4) Number of trials are fixed.

Coin toss probability:

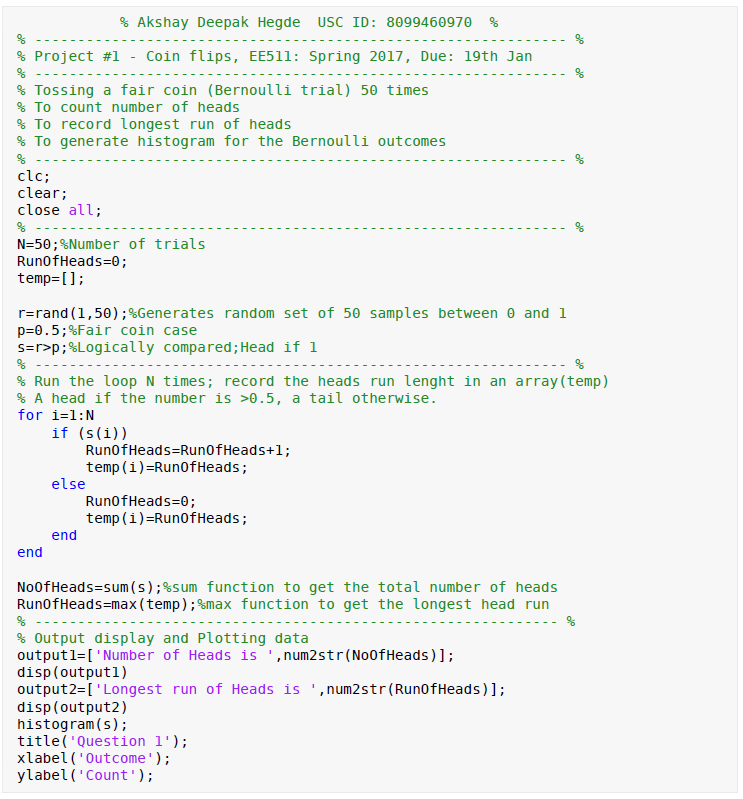
Coin toss is nothing but the experiment of tossing a coin. When we toss a coin, we can get either Head or Tail. Thus, the event of getting a head/tail is mutually exclusive. We can neither get a head and a tail simultaneously.

The sum of probability of getting a head and probability of getting a tail when a coin is tossed is always equal to 1.

Question 1: Simulate tossing a fair coin (a Bernoulli trial) 50 times.  Count the number of heads.  Record the longest run of heads.  Generate a histogram for the Bernoulli outcomes.

Description:

The problem demands the tossing of coin 50 times and to record the total number of heads, the longest run of heads and to generate a histogram for the outcomes. I will be using a inbuilt function of MATLAB, rand() to generate the random number in the range of 0 to 1. So rand(1,50) generates a set of 50 samples between 0 to 1. I will be setting p=0.5 as tossing of a fair coin is considered in the case (probability of getting a head is 50%). I will now logically compare the randomly generated 50 samples with p and store it in another array s. If the random number generated is greater than 0.5, we get a 1 which implies me a head in this case.

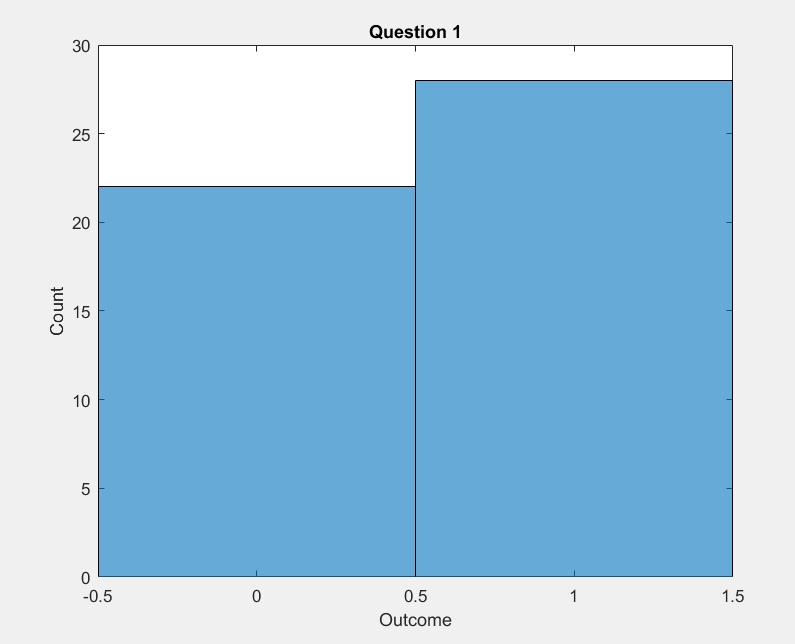


Sum() function will now give me the total number of heads in the 50 flips of the coin. To get the longest run of heads, I will loop through the flips 50 times. If it is a head, the run has been recorded in a counter else the counter is made 0 and the runs are being stored in a separate array. Max() function will now give me the maximum value of the array which implies the longest run of heads. A histogram graph is generated using the function histogram() and displayed using disp() function. The sample outputs and a histogram plot are included inline.

Output:

Number of Heads is 28

Longest run of Heads is 12



Comment:

The x-axis shows the possible outcomes which is either a head or a tail. The y-axis shows the frequency of each outcome. 28 heads and 22 tails are the counts out of 50 flips of coin in the particular case.

Question 1a.

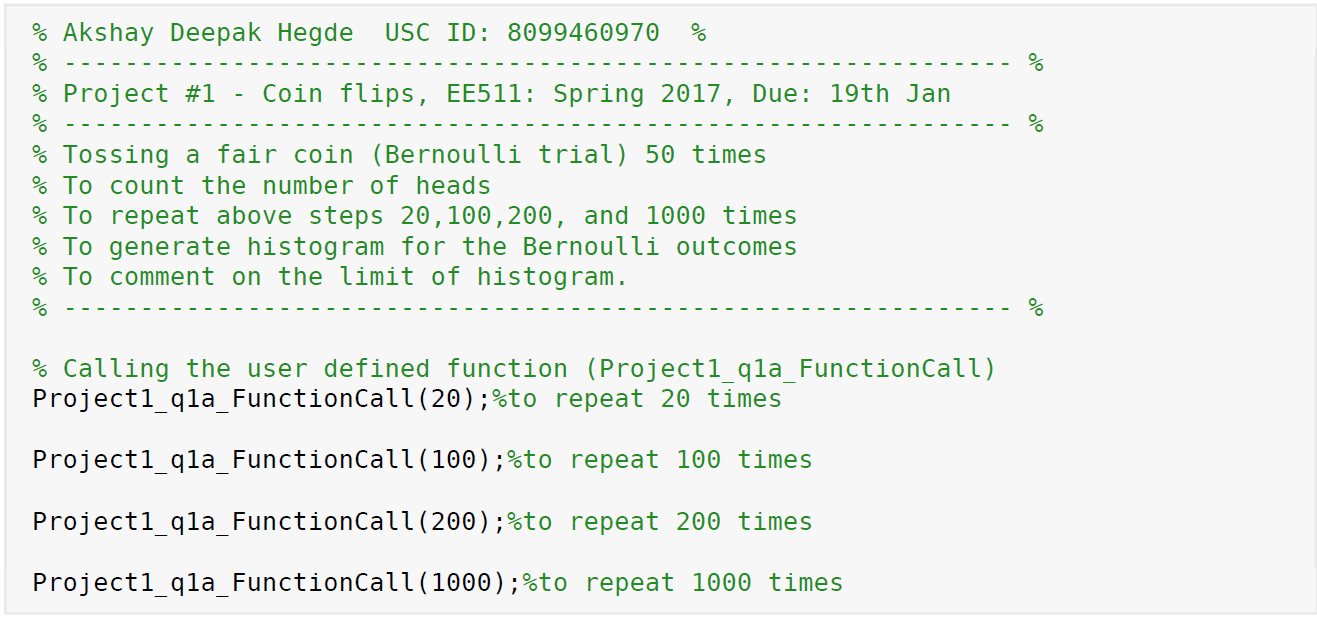
Repeat the above experiment 20, 100, 200, and 1000 times.  Generate a histogram for each showing the number of heads in 50 flips.  Comment on the limit of the histogram.

Description:

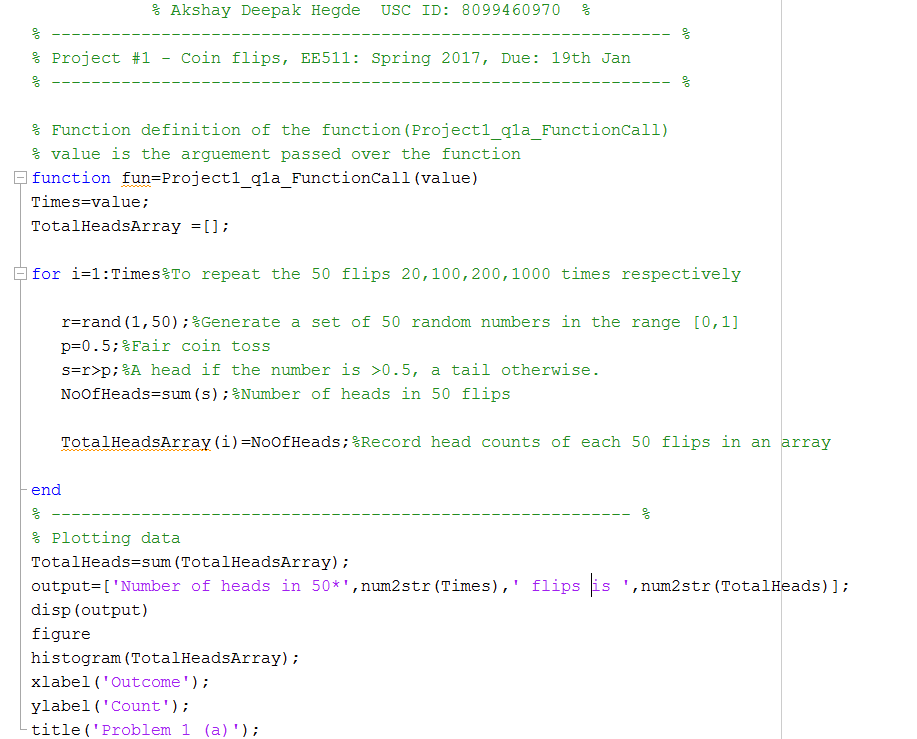
This is the extension of the above question where in the same experiment is repeated 20,100,200 and 1000 times respectively. A function call is made to repeat the same experiment different number of times.

The function is defined in a separate MATLAB file. The number of times the above experiment has to be performed is passed as a function argument and the value is stored in a variable called ‘times’. A loop is taken which runs for the particular times in each case which are 20,100,200, and 1000 respectively. In each run of the loop, a set 50 random numbers are generated using rand() and logically compared with 0.5, which is a fair trial condition. Number of heads in each 50 flips is counted using sum() function and is recorded in an array in each run of the for loop. Summation of this array when counted out of the loop gives total number of heads in each case. Output is displayed and the histogram plot is taken for each case. ‘figure’ command is used to get the histograms in separate windows.

Main code:



Function definition:



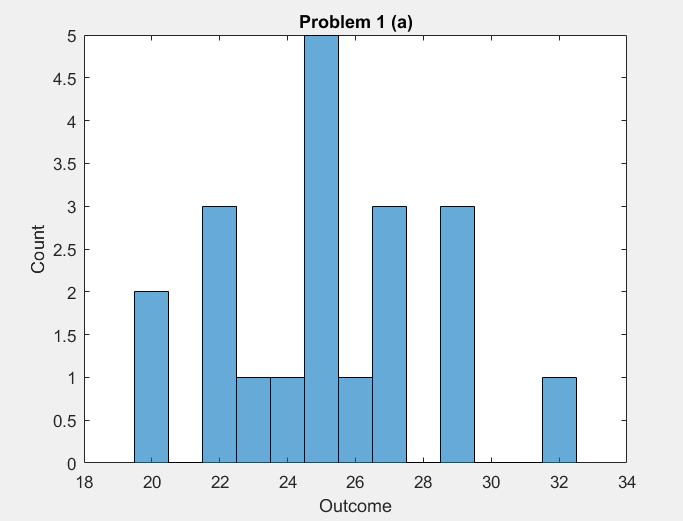
Output:

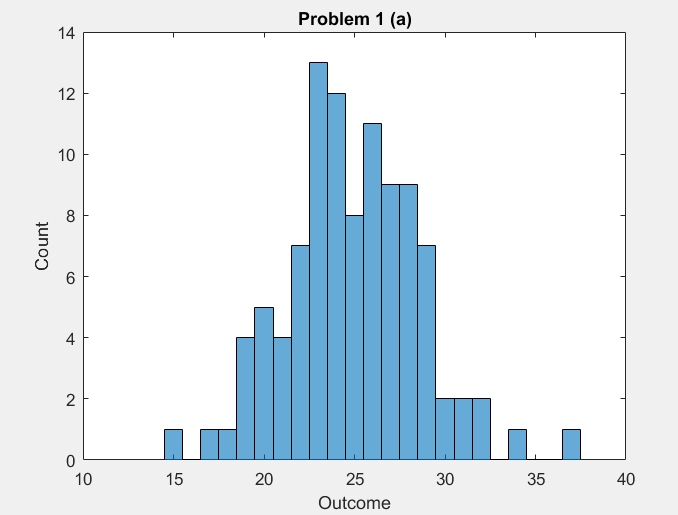
Number of heads in 50\*20 flips is 510

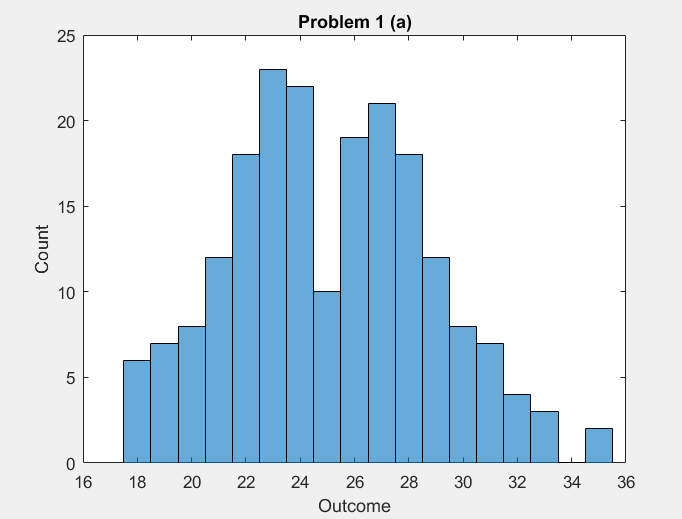
Number of heads in 50\*100 flips is 2491

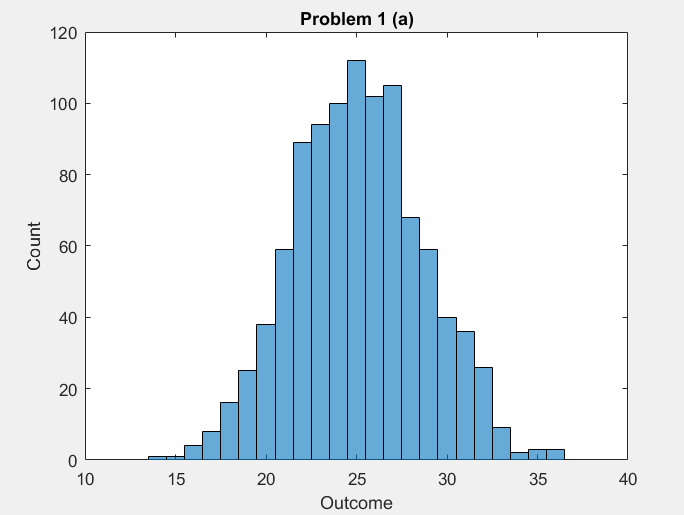
Number of heads in 50\*200 flips is 5124

Number of heads in 50\*1000 flips is 25136









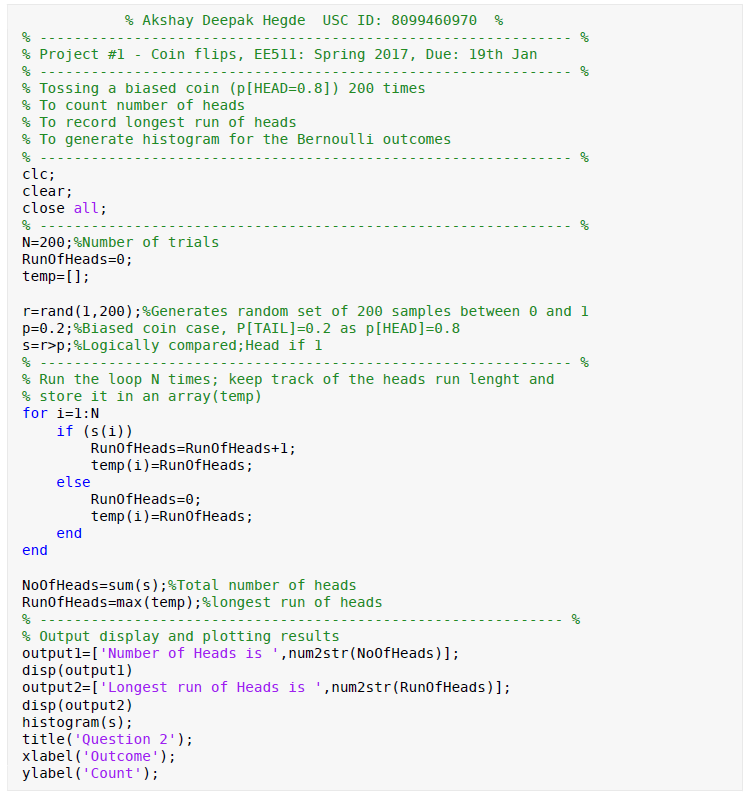
Comment:

The four histograms are for coin flips of 50\*20, 50\*100, 50\*200 and 50\*1000 respectively. The x-axis being the number of heads and the y-axis being the count. It is clearly evident from the above figures that, as N (number of times the experiment is repeated) increases the histogram is getting nice and smooth. In other words, as sample set of data grows, I am able to create more and more bins within the same range of values. I can see the histogram is getting thinner bins and the shape getting closer to a normal distribution. A normal distribution is a distribution of data where the data tends to be around a central value, a symmetry about the center with 50% of the data on the left and 50% on the right.

Question 2: Simulate tossing a biased coin 200 times where P[HEAD]=0.80. Count the number of heads.  Record thelongest run of heads.  Generate a histogram for the Bernoulli outcomes.

Description:

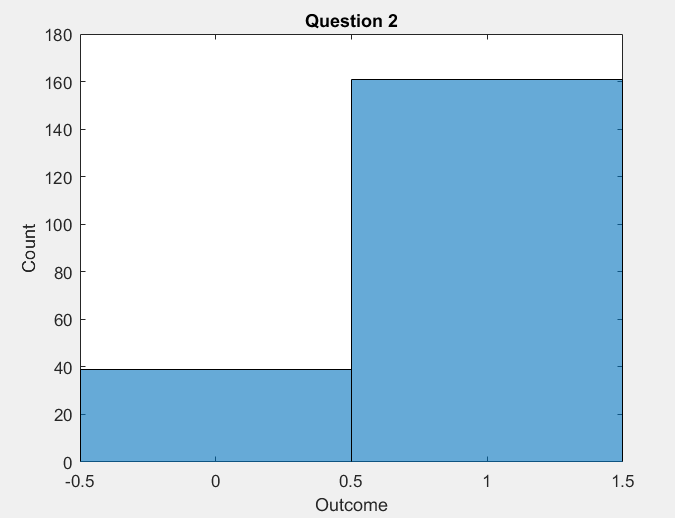
The experiment says the p[HEAD]=0.80 which means there is 80% chance to get a head when a coin is flipped; which means 20% chance of getting a tail. The number of trials to be conducted is 200 and hence a set of 200 random numbers is generated using rand() function. ‘p’ variable is set as 0.2 and the random numbers generated are logically compared with p. It is considered as a head if the comparison gives me a 1. Next, the loop is being run for 200 times and runs of heads is stored in an array. Sum() function gives me the total number of heads and the max() gives me the longest run of heads. Finally, a outputs are displayed and histogram is plotted.



Output:

Number of Heads is 161

Longest run of Heads is 14



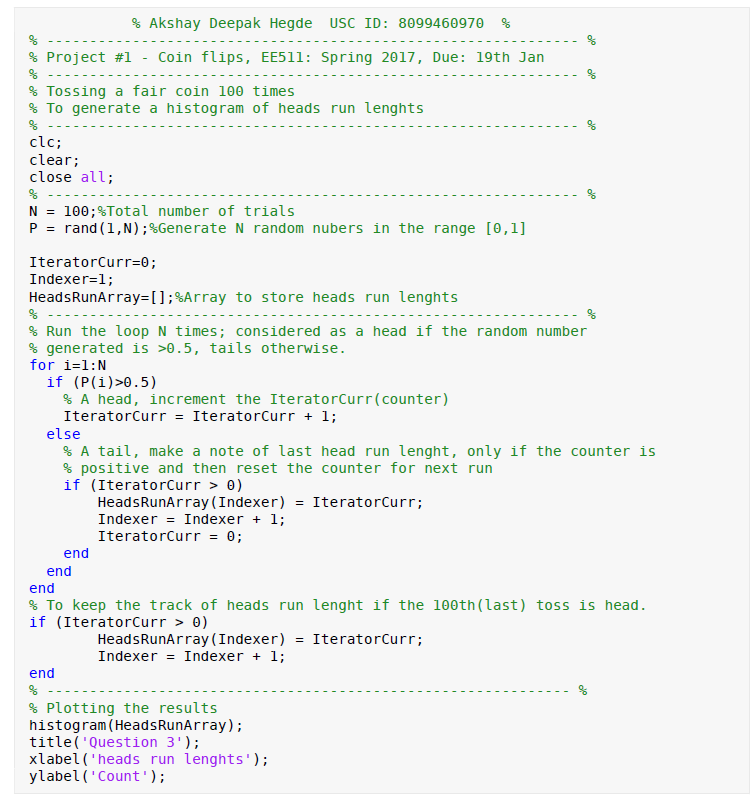
Comment:

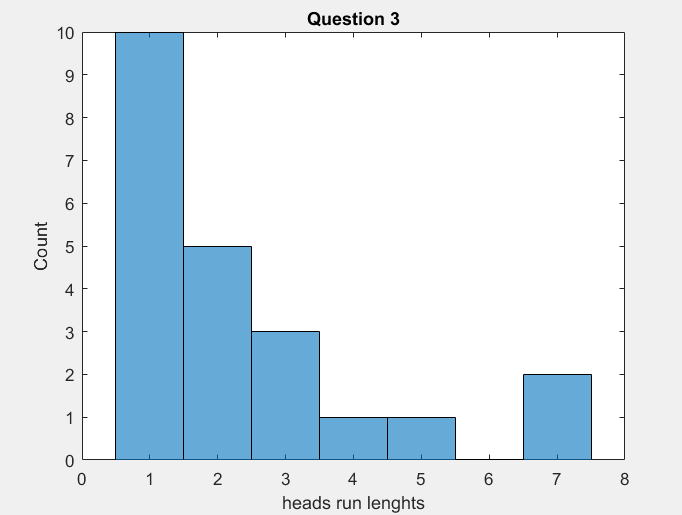
In the above histogram, x-axis represents the outcomes which is either a head or a tail and y-axis represents the count of number of heads and tails. As seen, out of 200, I am getting 161 heads since the p[HEAD]=0.8.

Question 3: Simulate tossing a fair coin 100 times.  Generate a histogram showing the heads run lengths.

Description:

For the above experiment, 100 random numbers are generated using rand() function and the loop is run for 100 times. In the loop, if the random number generated is greater than 0.5(considered as head) a counter is set. The counter is incremented if the next flip is also head or else the counter is checked. The counter is greater than 0 means the counter must be stored in an array and the counter is cleared for the next run of the loop. If the counter is 0, that means the previous flip was also a tail and it is ignored. To keep the track of 100th toss, a check is made if the counter is greater is 0. If it is, the value is stored in the array. At last, the histogram of the array gives the required plot.





Comment:

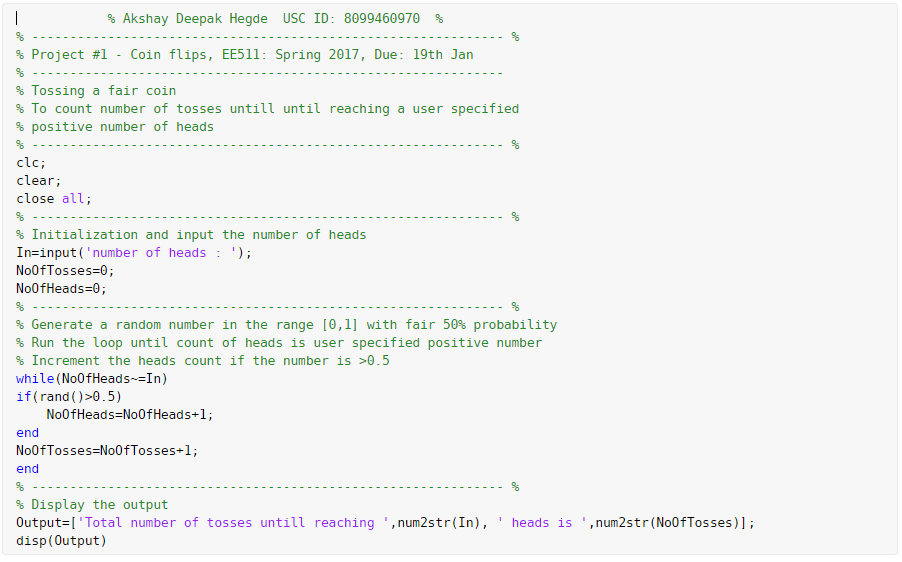
In the above histogram, x-axis shows the run lengths of heads and y-axis the count.

Question 4:

Simulate tossing a fair coin and count the number of tosses until reaching a user‐specified positive number of heads

Description:

The experiment demands the tossing a fair coin until the user specified positive number is reached. Input() function is used to prompt the user to input the value. A while loop is being run. In the loop, the random number is generated using rand() function and a count of number of heads is being tracked if the random number generated greater than 0.5(considered as a head). Track of total number of tosses is also kept out of the loop and the while loop is being run until the total heads is equal to the user specified number. Finally, the output is displayed using disp() function and the num2str() function which converts a number to a string.



Output:

number of heads : 20

Total number of tosses untill reaching 20 heads is 43