

University of Southern California

EE511

Simulation Methods for
Stochastic Systems

Project #1 - Coin Flips

BY

Mohan Krishna Thota

USC ID: 6683486728

mthota@usc.edu

Theory:

Bernoulli trials:

Bernoulli trial is a kind of an event when only two outcomes are possible. Like for an instance, flipping a coin is a Bernoulli event since only two outcomes are possible i.e., heads and tails

Therefore, the probability of success is p . then the probability of failure is $1-p$.

The characteristics of a Bernoulli trial are:

- 1) Independent trials i.e., the outcome of an event does not influence the other
- 2) Probability of both outcomes remains same through each trial.
- 3) Only two outcomes i.e., either success or failure.

Probability of coin toss:

Coin toss probability is nothing but the probability of getting a head or toss which are mutually exclusive which means one event does not influence the other one

As we are aware that the sum of the probability of all the events are equal to one. Here there are only two events therefore if the probability of getting a head is p then the other event i.e., getting tails is $1-p$.

Question1:

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:

- Here we need to toss a fair coin 50 times and the number of heads out of the entire tosses are calculated.
- Also the longest run of heads i.e., the continuous heads without any other event in between. Histograms are generated for the Bernoulli outcomes.
- Here in the program written an inbuilt function of MATLAB 'rand' is used to generate 50 sample values between 0 and 1.
- Let the value of p be 0.5 so that it can be compared with the randomly generated sample in order to differential it whether it is a head or a tail i.e., head if the sample is greater than p and tails when $p < 0.5$.
- If head, then the result is stored in an array and the values are counted using the sum() function in order to find the number of heads.
- In order to get the longest run of heads, a loop is given for all the flips of the coin, if a head event occurs then count value is incremented by 1 and so on and whenever a tail event occurs then the count value is made equal to zero.
- In this way we can get the longest run of the heads by using max() function of the count value
- 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.

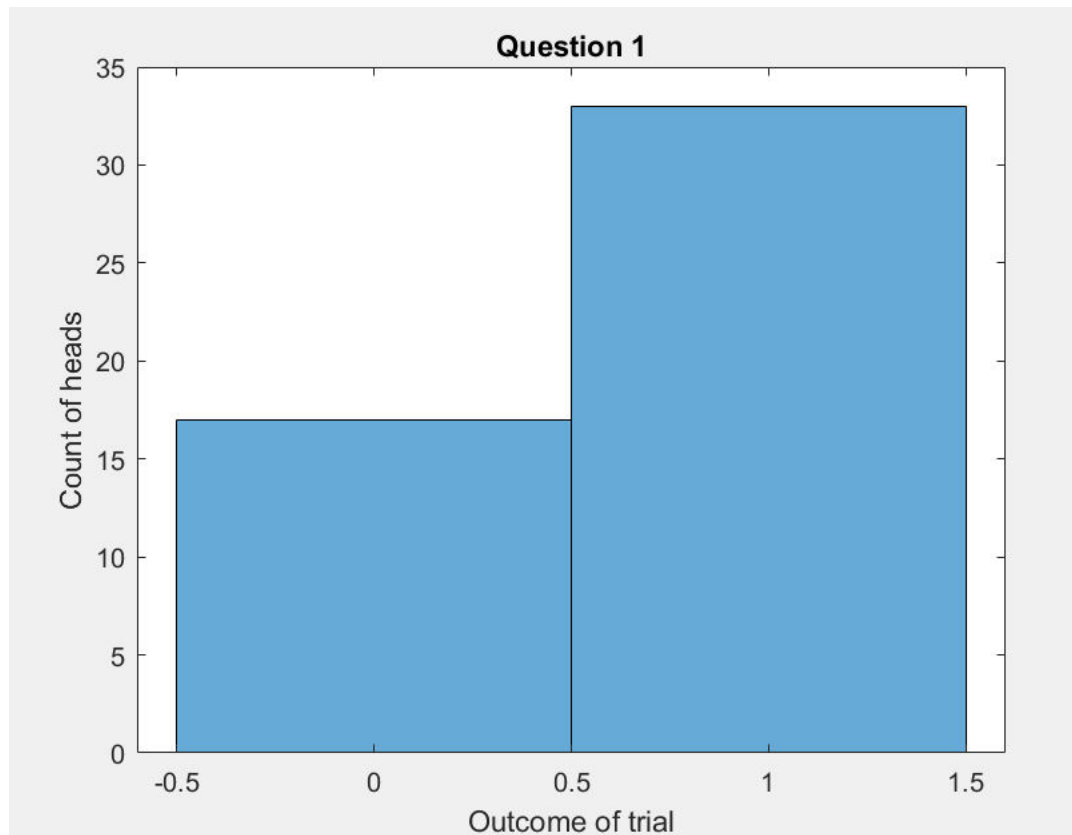
Matlab programme:

```
clc;
clear;
close all;
Head_run=0;
temp_array=[];
Number_of_trials=50; %number of trials
random_samples=rand(1,50); %50 random samples are generated between 0 and 1
probability_of_outcome=0.5; %fair coin tossing
logical_conditioning=random_samples>probability_of_outcome; %logical condition
k=0;
for i=1:Number_of_trials
    if(logical_conditioning(i))
        Head_run=Head_run+1;
        temp_array(i)=Head_run;
    else
        Head_run=0;
        temp_array(i)=Head_run;
    end
end
head_count=sum(logical_conditioning);
Longest_run_Heads=max(temp_array);
output1=['the number of occurrences of head is ',num2str(head_count)];
disp(output1)
output2=['the Longest run of heads is ',num2str(Longest_run_Heads)];
disp(output2)
histogram(logical_conditioning); %Histogram generation for the bernoulli outcomes
title('Question 1');
xlabel('Outcome of trial');
ylabel('Count of heads');
```

Output:

the number of occurrences of head is 22

the Longest run of Heads is 4



Comment:

- The histogram shows us the graphical representation of the occurrences of the events for the 50 flips of the coin.
- Here as we know 'p' is on the X- axis and the number of the occurrences of the head or on Y-axis.

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:

- Here it is same as the above experiment except for that the coin is tossed 20,100,200,1000 times differently.
- Function call is used to perform the experiment different times
- The function is defined in a separate MATLAB file.
- The number of the tosses of the experiment is given as a function argument by variable named times which determines the number of occurrences of the event.
- For each loop, in the program written an inbuilt function of MATLAB 'rand' is used to generate 50 sample values between 0 and 1.

- Let the value of p be 0.5 so that it can be compared with the randomly generated sample in order to differential it whether it is a head or a tail i.e., head if the sample is greater than p and tails when $p < 0.5$.
- Sum() function is used in each loop to count the number of occurrences of the head and the values are stored in an array.
- 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 Function=HW1_Q1a_FunctionCall(times)
Times=times;
head_array=[];

for i=1:Times%To repeat the 50 flips 20,100,200,1000 times respectively

    random_value=rand(1,50);%Generate a set of 50 random numbers in the range [0,1]
    probability=0.5;%Fair coin toss
    s=random_value>probability;%A head if the number is >0.5, a tail otherwise.
    head_count=sum(s);%Number of heads in 50 flips

    head_array(i)=head_count;%Record head counts of each 50 flips in an array

end
% ----- %
% Plotting data
head_total=sum(head_array);
output=['Number of heads in 50*',num2str(Times),' flips is ',num2str(head_total)];
disp(output)
figure
histogram(head_array);
xlabel('Outcome');
ylabel('Count');
title('Problem 1 (a)');
```

```
HW1_Q1a_FunctionCall(20);%to repeat 20 times
```

```
HW1_Q1a_FunctionCall(100);%to repeat 100 times
```

```
HW1_Q1a_FunctionCall(200);%to repeat 200 times
```

```
HW1_Q1a_FunctionCall(1000);%to repeat 1000 times
```

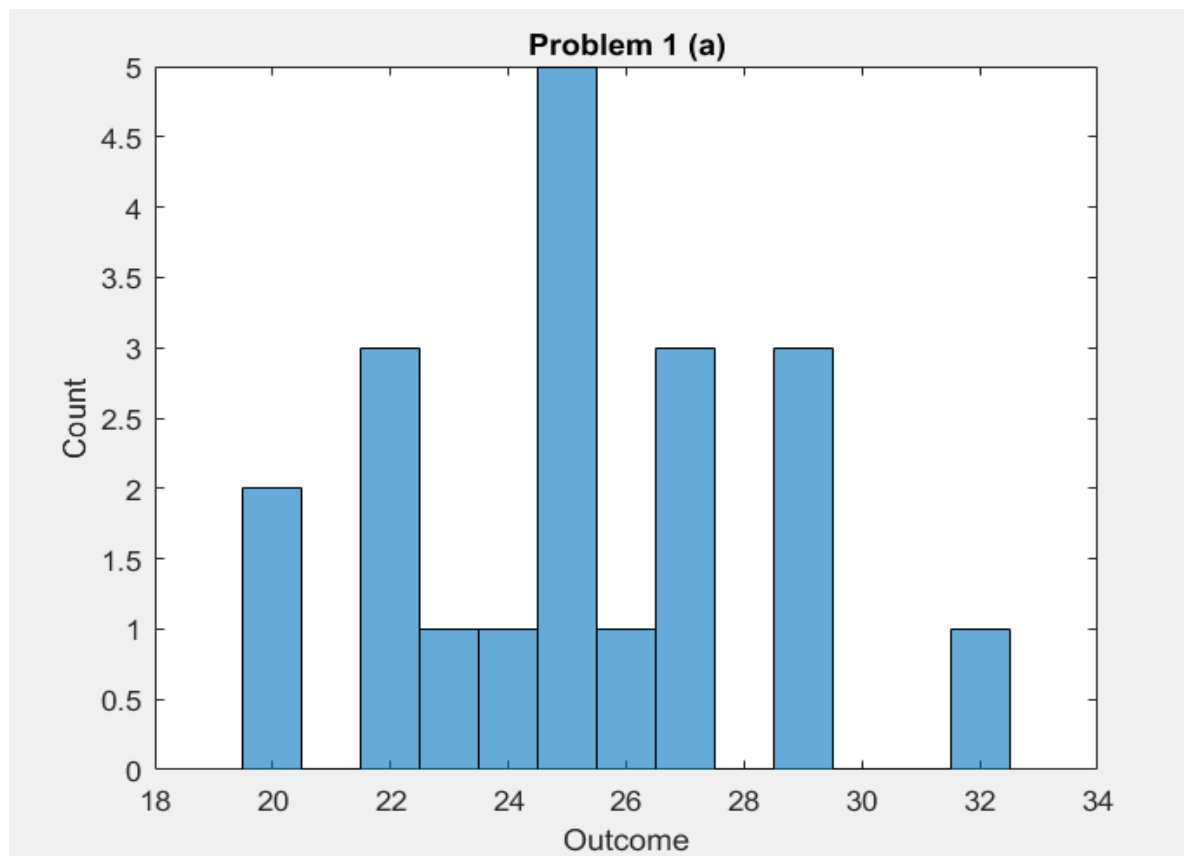
Output:

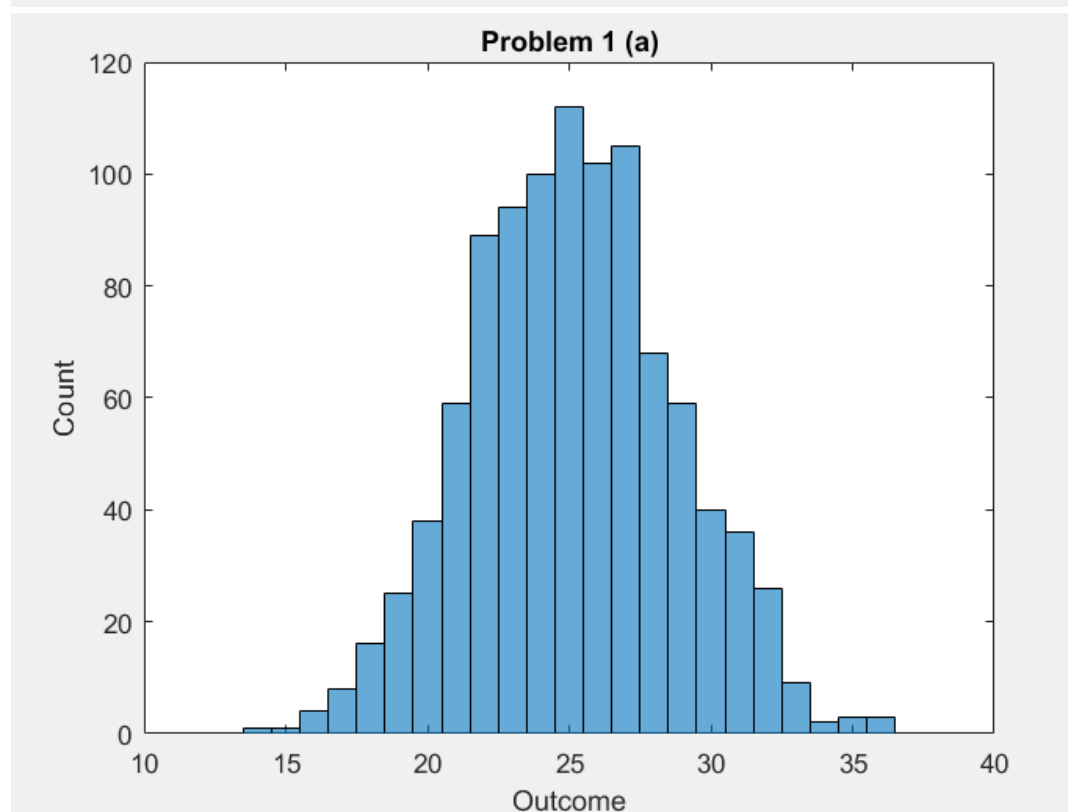
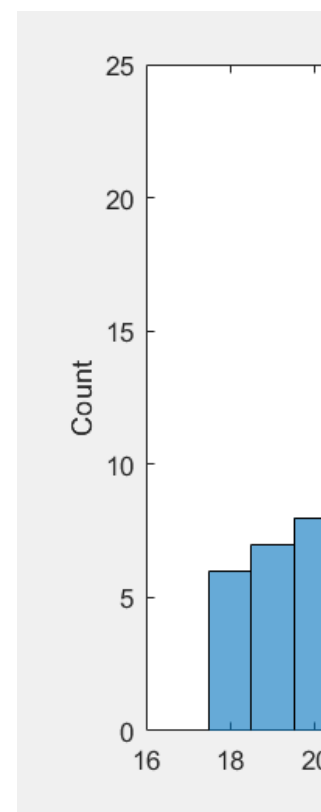
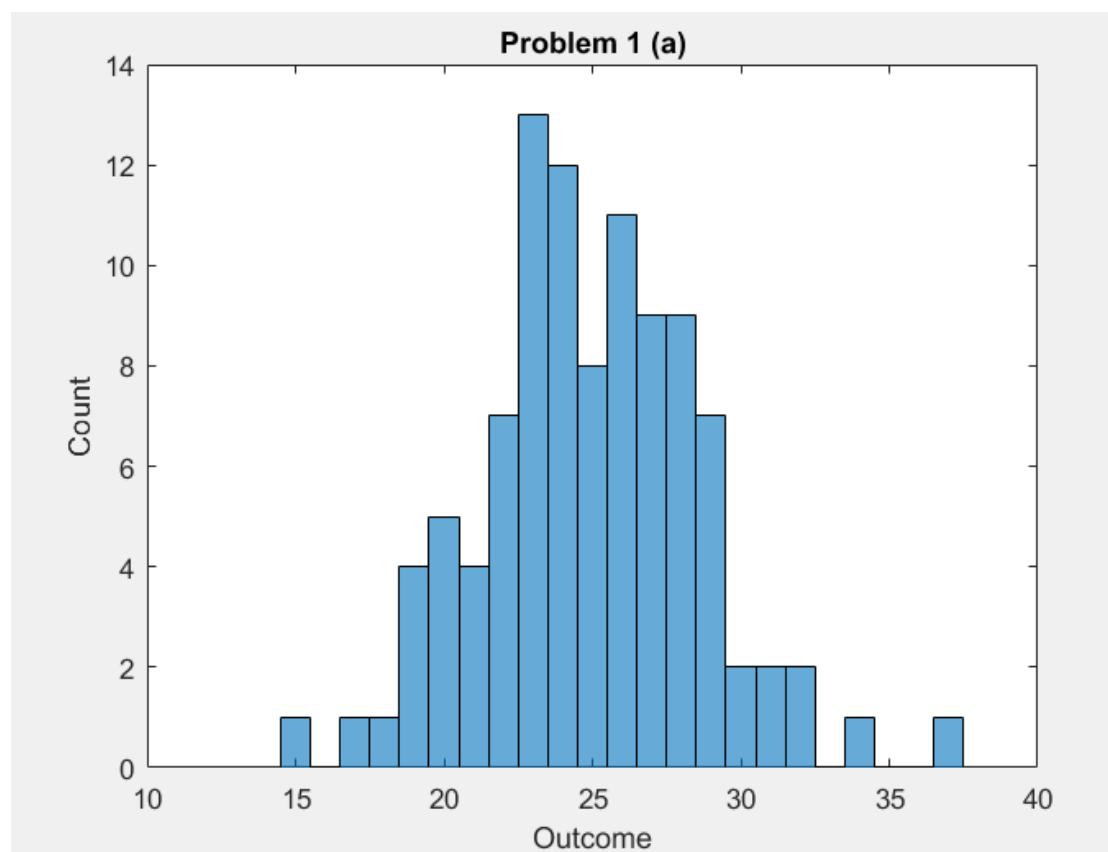
Number of heads in 50*20 flips is 510

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:

- Histograms above are the outputs of the coin flipped 20,100,200,1000 times respectively.
- Number of heads is given by the x axis
- From the above figures we can see that the Histogram is getting smooth with the increase in N i.e., number of flips of the coin.
- More number of bins are created with the same range of the number of heads
- As a result of which the bins are becoming thinner making it 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 $P[\text{HEAD}] = 0.80$. Count the number of heads. Record the longest run of heads. Generate a histogram for the Bernoulli outcomes.

Description:

- As per the question, the coin has to be tossed for a total of 200 times.
- It is given in the question that $P[\text{HEAD}] = 0.8$ which means the probability of getting a head is more than that of the probability of getting a tail which is 0.2
- Since the number of trials is 200, `rand()` function is used to generate 200 random values between 0 and 1.

- If head, then the result is stored in an array and the values are counted using the sum() function in order to find the number of heads.
- In order to get the longest run of heads, a loop is given for all the flips of the coin, if a head event occurs then count value is incremented by 1 and so on and whenever a tail event occurs then the count value is made equal to zero.
- In this way we can get the longest run of the heads by using max() function of the count value
- I will loop through the flips 200 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.

```

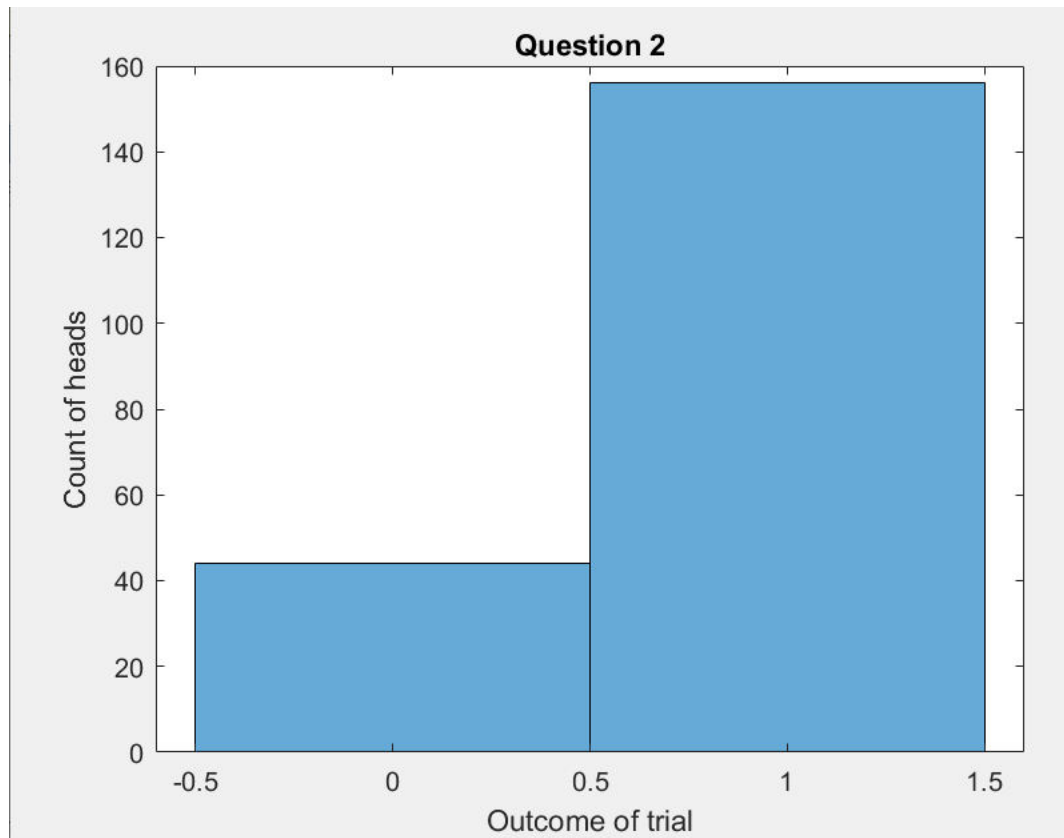
clc;
clear;
close all;
Head_run=0;
temp_array=[];
Number_of_trials=200; %number of trials
random_samples=rand(1,200); %50 random samples are generated between 0 and 1
probability_of_outcome=0.2; %fair coin tossing
logical_conditioning=random_samples>probability_of_outcome; %logical condition
k=0;
for i=1:Number_of_trials
    if(logical_conditioning(i))
        Head_run=Head_run+1;
        temp_array(i)=Head_run;
    else
        Head_run=0;
        temp_array(i)=Head_run;
    end
end
head_count=sum(logical_conditioning);
Longest_run_Heads=max(temp_array);
output1=['the number of occurrences of head is ',num2str(head_count)];
disp(output1)
output2=['the Longest run of heads is ',num2str(Longest_run_Heads)];
disp(output2)
histogram(logical_conditioning); %Histogram generation for the bernoulli outcomes
title('Question 2');
xlabel('Outcome of trial');
ylabel('Count of heads');

```

Output:

Number of Heads is 168

Longest run of Heads is 18



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 $p[\text{HEAD}] = 0.8$.

Question 3:

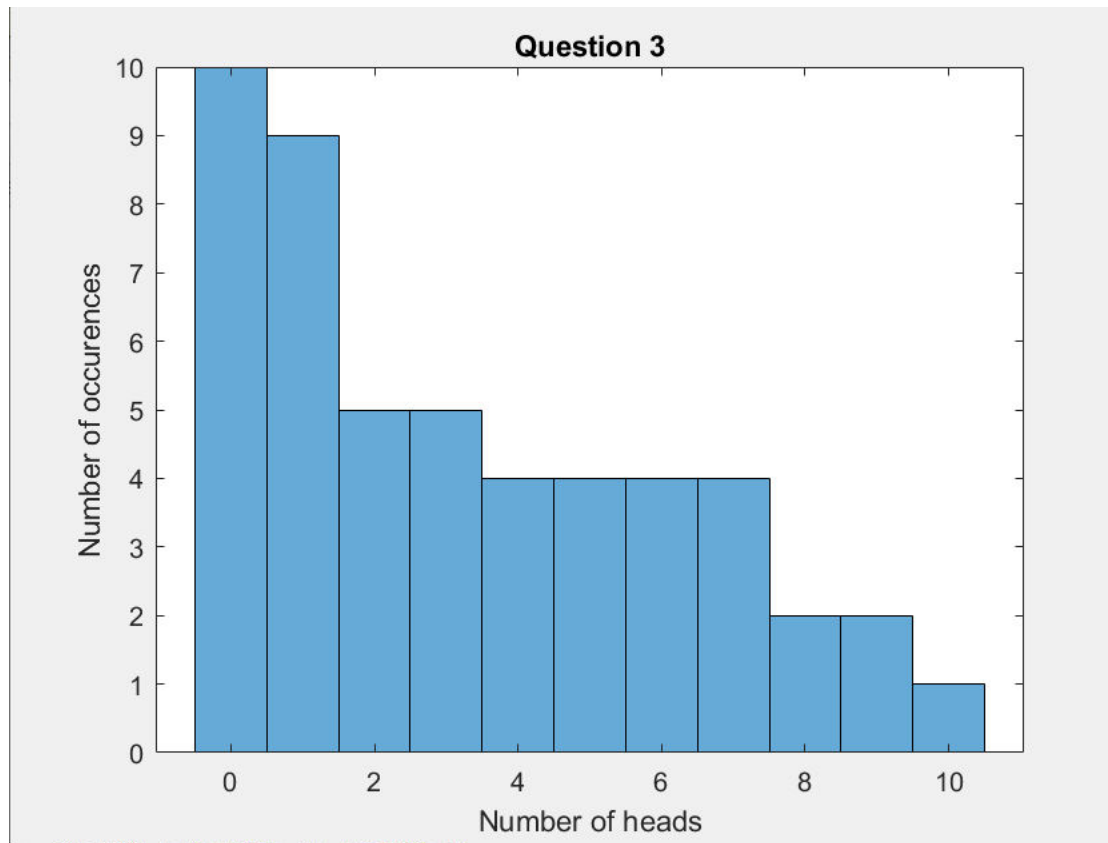
Simulate tossing a fair coin 100 times. Generate a histogram showing the heads run lengths.

Description:

- Here a fair coin is tossed 100 times.
- An inbuilt function of matlab `rand()` is used to generate random values between 0 and 1.
- The randomly generated value is compared against the 0.5 to verify whether the output obtained is head or tail.
- And the counter is incremented by one for every random value encountered greater than 0.5.
- An array is used to store the counter value only when the counter value is one or more.
- 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.

- 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.

```
clc;
clear;
close all;
Head_run=0;
temp_array=[];
Number_of_trials=100; %number of trials
random_samples=rand(1,100); %50 random samples are generated between 0 and 1
probability_of_outcome=0.2; %fair coin tossing
logical_conditioning=random_samples>probability_of_outcome; %logical condition
k=0;
for i=1:Number_of_trials
    if(logical_conditioning(i))
        Head_run=Head_run+1;
        temp_array(i)=Head_run;
    else
        Head_run=0;
        temp_array(i)=Head_run;
    end
end
head_count=sum(logical_conditioning);
output1=['the number of occurrences of head is ',num2str(head_count)];
disp(output1)
histogram(temp_array); %Histogram generation for the bernoulli outcomes
title('Question 3');
xlabel('Number of heads');
ylabel('Number of occurrences');
```



Comment:

In the above histogram, x-axis shows the counts of heads and y-axis the Number of occurrences.

Question 4:

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

Description:

- Here from the question it is asked that a coin is tossed until a user specified number of heads is reached.
- An inbuilt function `input()` is used to get the user input
- A while loop is used to verify whether the number of heads obtained is less than the user given input
- 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.
- After this an inbuilt `rand()` function is used to generate a random number between 0 and 1 and is verified opposite the 0.5 to check whether the obtained number is heads or

tails i.e., if the random value is greater than p then it is heads and if it is less than p then it is tails

- Finally, the output is displayed using disp() function and the num2str() function which converts a number to a string.

```
1 - -----question4-----
2 - clc;
3 - clear;
4 - close all;
5 - %input initialisation
6 - head_input=input('Number of heads: ');
7 - Head_count=0;
8 - toss_count=0;
9 - %logical conditioning
10 - while(Head_count~=head_input)%where the number of heads obtained is less than the user input
11 -     if(rand())>0.5
12 -         Head_count=Head_count+1;
13 -     end
14 -     toss_count=toss_count+1;
15 - end
16 - Output=['Total number of tosses untill reaching ',num2str(head_input), ' heads is ',num2str(toss_count)];
17 - disp(Output)
```

Command Window

```
Number of heads: 5
Total number of tosses untill reaching 5 heads is 11
```

Output:

Number of heads : 5

Total number of tosses until reaching 5 heads is 11