

```

%%-----
%%Project-1:: Question - 2
%%To Simulate a biased coin(p[head]=0.8) toss 200 times.Count the number of
%%Heads, record the longest run and generate a histogram for the Bernoulli outcomes

%%The below function performs the following
%%1) Rand function(Uniformly distributed random numbers) to generate a random
    %%number between (0,1) as per user defined num_of_flips
%%2) find function to get the indices of above vector having '0'(Heads) and
    %%dump it in an array
%%3) Add an elements with 1 to above array to check if the result equals
    %%the next element(consecutive heads) and build Heads distribution
%%4) Now that we have an array of consecutive heads distribution, plot
    %%histogram for those elements and get the Max of that array.
%%-----
function [] = coin_toss(num_of_flips)
% generate a vector of num_of_flips Flips with p[Head] = 0.8
coin_flips = rand(num_of_flips, 1) > 0.8;
% get the indices of heads(0) from the overall coin_flips array
Indices_of_heads = find(coin_flips==0);
[len_of_indices_vector,~] = size(Indices_of_heads);
consecutive_heads=1;

[num_of_heads,~] = size(Indices_of_heads);
disp('Number of Heads')
disp(num_of_heads);
head_run_lengths = zeros(1, num_of_flips);
%For loop to get the number of instances of consecutive heads
for iter=1:(len_of_indices_vector-1)
    if iter == (len_of_indices_vector-1)
        head_run_lengths(iter+1) = consecutive_heads;
    end
    if (Indices_of_heads(iter) + 1) ~= Indices_of_heads(iter+1)
        head_run_lengths(iter) = consecutive_heads;
        consecutive_heads=1;
    else
        consecutive_heads = consecutive_heads+1;
    end
end
%removes the '0's in the array containing distribution of heads
head_run_lengths = head_run_lengths(head_run_lengths~=0);

disp('Distribution of consecutive heads')
disp(head_run_lengths)
%Gets the longest instance of consecutive heads
Max_heads_run_length = max(head_run_lengths);
disp('Longest consecutive run of heads')
disp(Max_heads_run_length)
histogram(coin_flips);

```