Binomial Random Variable

```
n_trials = 1000;
n_flips = 21;

values = zeros(1, n_flips);
valuesFloat = zeros(1, n_flips);

% Simulating coin flips
for i = 1:n_trials
    numberOfHeads = sum(randi([0,1], 1, n_flips)); % 
Vectorized operation
    values(numberOfHeads + 1) = values(numberOfHeads 
+ 1) + 1; % Adjusting index since MATLAB arrays are 
1-indexed
end

valuesFloat = values / n_trials; % Vectorized 
operation

% Plotting
figure;

% Integer Values
subplot(2,1,1);
bar(1:n_flips, values);
xlabel('Number of Heads');
ylabel('Frequency');
title('Integer Values');
grid on;

% Float Values
subplot(2,1,2);
bar(1:n_flips, valuesFloat);
xlabel('Number of Heads');
ylabel('Frequency');
title('Float Values');
grid on;
```

Geometric Random Variable

```
totalExperiment = 1000;
numberOfFlipsUntilFirstHead = zeros(1, 20);

for i = 1:totalExperiment
    numberOfTrial = 0;
    while randi([0,1]) == 0
        numberOfTrial = numberOfTrial + 1;
    end
    numberOfFlipsUntilFirstHead(numberOfTrial + 1) =
numberOfFlipsUntilFirstHead(numberOfTrial + 1) + 1;
end

bar(numberOfFlipsUntilFirstHead);
xlabel('Number of Flips to get First Head');
ylabel('Frequency');
title('Geometric Random Variable');
grid on;
```

Steady State Matrix Generation

```
% Example matrix
matrix = [1, 0, 0;
          0.002, 0.998, 0;
          0, 0.002, 0.998];

% Parameters
threshold = 1e-5;
maxIterations = 10000;

result = matrix;

for i = 1:maxIterations
    previousResult = result;
    result = result * matrix;
    disp(i);
    if norm(result - previousResult, 1) < threshold
        break;
    end
end

roundedMatrix = round(result, 2);

disp('Steady State Matrix:');
disp(result);
disp('Rounded Steady State Matrix:');
disp(roundedMatrix);
```