CS100M Lab Exercise 6

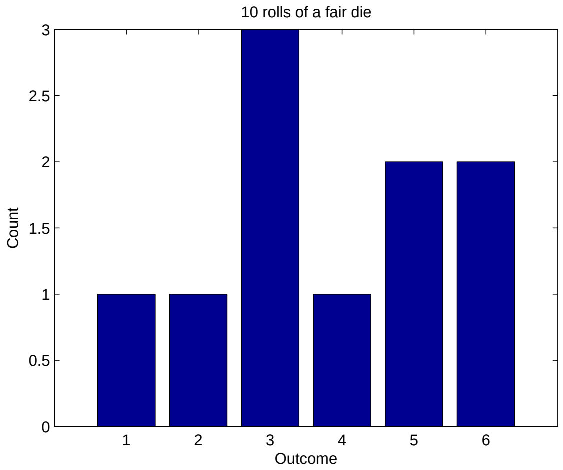
1 Different ways to create vectors

Type the following expressions in the MATLAB *Command Window* to see what kind of vectors they create. Write the resulting vectors (and answer the questions) on the blanks.

```
a= zeros(1,4)  %________________

b= zeros(4,1)  %________________  What do the arguments specify?________________________

c= ones(1,3)   %________________

d= 10:2:17     %________________

f= 10:-1:17    %________________

g= [10 20 40]  %________________  What does the space separator do?_____________________

h= [10,20,40]  %________________  What does the comma separator do?_____________________

k= [10;20;40]  %________________  What does the semi-colon separator do?________________

m= [a g]       %________________

n= [b;k]       %________________

p= [a k]       %ERROR--mismatched dimensions! (Attempt to concatenate a column to a row)

q= b'          %________________  This operation is called "transpose"

r= [a b']      %________________
```

2 Roll a fair die

Write a function `rollDie(n)` to simulate the rolling of one six-sided die `n` times. The function draws a histogram of the result. Below is an example histogram. Use a vector to count the number of times that each face occurs.

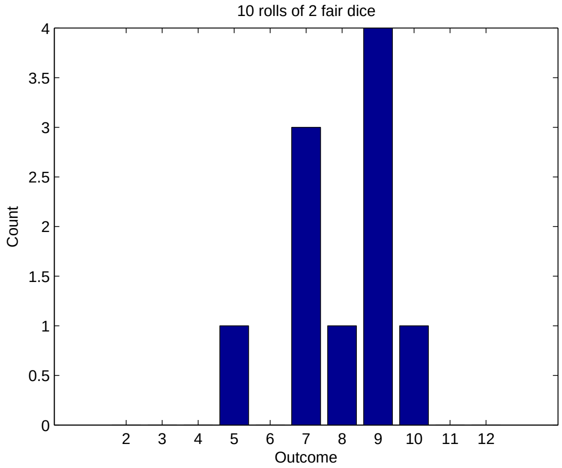


For example, use a vector `count` with six cells such that `count(f)` is the number of times that face `f` has occured. Solve this problem without using an `if` statement. You will need to generate random integers in [1..6].

Use the built-in function bar to draw a bar graph: bar(1:6, count) where the first argument (1:6) contains the labels for the bars and the second argument (count) contains the counts for the six possible outcomes. Label the axes and the figure.

3 Roll multiple dice

Write a function rollDice(n,d) to simulate the rolling of d six-sided dice n times. We define the *outcome* of rolling d dice once to be the sum of the faces that show up. In the function, create a vector count such that count(c) is the number of times that outcome c has occured. *Do not* use built-in function sum. Your function draws a histogram of the result. Below is an example histogram for small n. What shape do you expect to see for large n?



4 Examining a subarray

Write a function vectorQuery(v,n,r) to determine whether the number r appears in the first n cells of vector v. The function returns 1 if r is in the first n cells of v and 0 otherwise. Your function assumes that v is a vector of numbers, n is a positive integer, and r is a number. Use a loop to do the search. Make sure that the loop index doesn't go “out of bounds” (if n is greater than the length of vector v).

5 Concatenating arrays

Write a function sequence(m) that generates a sequence of random *integer* numbers between 1 and m, inclusive, stopping when a value is repeated for the first time. The function returns an array containing all the numbers generated (in the order in which they were generated) except for the last value that is a repeated occurrence.

Example: If the generated sequence is 3 1 9 5 7 2 5, the array to be returned should be 3 1 9 5 7 2.

Hints: 1) Use the function vectorQuery that you have developed already. 2) The symbol for the empty array is []. When “building” an array, the space or comma separator puts two items side by side—creates a row. Below is an example:

```
v= [];       % v is empty array
v= [v 7];    % now v is [7]
v= [v -5];   % now v is [7, -5]
```

Please delete your files from the computer before you leave the lab.