ECS174 PS0

Haley Sanders - Turner

April 2017

1 Problem 2

a. Randperm() returns a vector of random permutations of the range consisting of 1 to the input value. Randperm(500) is assigned to the variable x, without printing out the result due to the ';' at the end of the statement.

b. The first command creates a 3 x 3 matrix and assigns it to the variable a. [1,2,3] sets the values of the first elements of the columns as 1,2 and 3 respectively, while [4 5 6; 7 8 9] creates two rows. The second command assigns to the variable b the whole of the third column.

c. The first command does the same as the first command for part b. The second command assigns to b the entire matrix that was assigned to a.

d. Randn(m, n) generates a matrix of n columns with m rows. The values within it are taken from the normal distribution. ii. The second command finds all of the values within f that are positive but not zero. These values are put into a column vector.
e.The first command creates a row vector with five elements (all initially set to 0) and then adds 0.5 to each of the elements.The second command uses element by element multiplication to multiply each value of the row vector of 1's created by the ones() call by 0.5 . The third command does element by element addition of the row vectors x and y.

f. The first command creates a row vector of values ranging from 1 to 50.The second command creates a row vector of values ranging from 50 to 1 by starting at the end of the vector named a and "stepping" backwards by 1 each time.

2 Problem 3

Listing 1: Random 6-Sided Die Roll

```
function roll = random_roll()
roll = floor(1 + (6 - 1).*rand(1));
end
```

Listing 2: Reshaping Y

```
function reshape_y()

y = [1 2 3 4 5 6]';
Z = reshape(y,2,3)
end
```

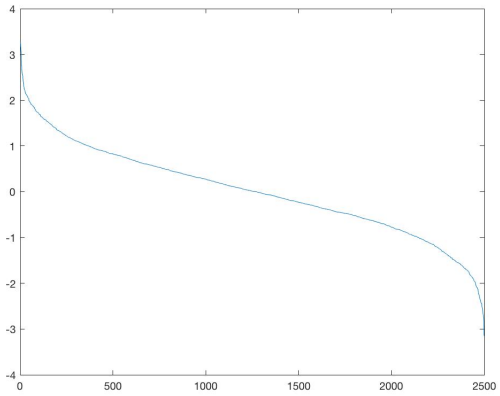
Listing 3: Maximum of matrix Z

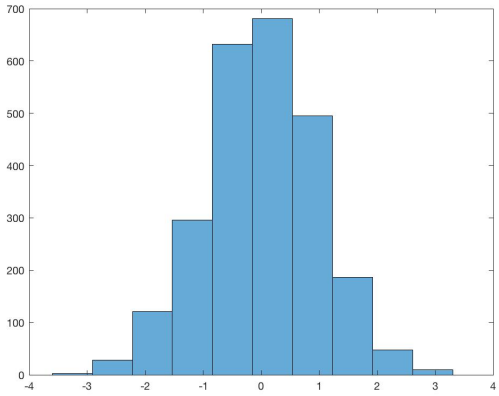
```
function Z_max()
 Z = reshape_y();
 x = max(max(Z));
 [r,c] = find(Z == x);
 x
 r
 c
end
```

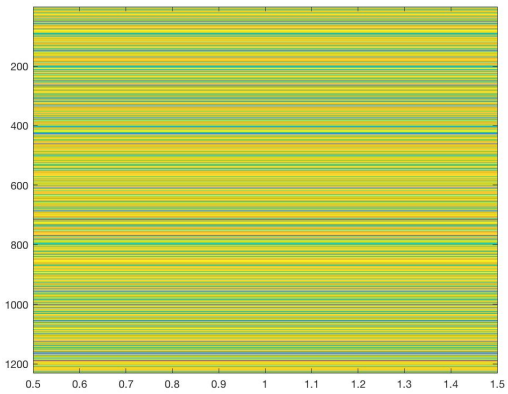
Listing 4: Counting 8's

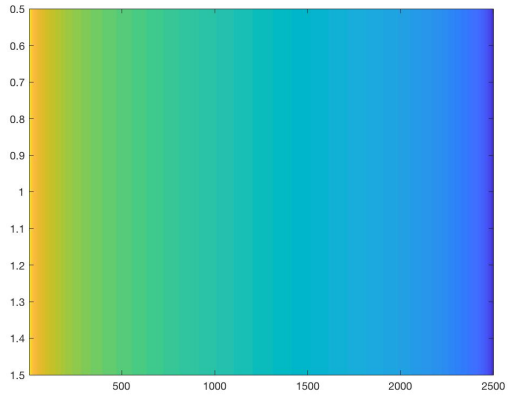
```
function roll = random_roll()
roll = floor(1 + (6 - 1).*rand(1));
end
```

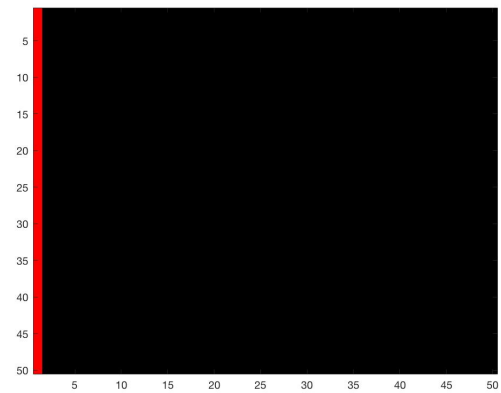
3 Problem 4


4
3
2
1
0
-1
-2
-3
-4
0
500
1000
1500
2000
2500


700
600
500
400
300
200
100
0
-4
-3
-2
-1
0
1
2
3
4


200
400
600
800
1000
1200
0.5
0.6
0.7
0.8
0.9
1
1.1
1.2
1.3
1.4
1.5


0.5
0.6
0.7
0.8
0.9
1
1.1
1.2
1.3
1.4
1.5
500
1000
1500
2000
2500


5
10
15
20
25
30
35
40
45
50
5
10
15
20
25
30
35
40
45
50