

Problem Set 0

Part I.

2. A. `x = randperm(100);` The `randperm` function, when passed one positive integer `N`, generates a random permutation of the integers `1,2,3...N`. In this case, a random permutation of the integers `1...100` is generated and assigned to the variable `x`.
- B. `>> a = [1,2,3; 4 5 6; 7 8 9]; >> b = a(3,:);` The first command creates a `3x3` matrix and assigns it to the variable “`a`” such that the comma separated numbers are entries in a row, and each row is demarcated by a semicolon. The second assigns the submatrix of “`a`” to “`b`” containing the former’s third row and every column.
- C. `>> a = [1,2,3; 4 5 6; 7 8 9]; >> b = a(:);` The column vector generated from the entries of “`a`” is assigned to “`b`.”
- D. `>> f = randn(4,1); >> g = f(find(f > 0));` The first command generates a `4x1` matrix of normally distributed (mean = 0, standard deviation = 1) pseudorandom numbers and assigns it to “`f`.” The second is a function composition; first it determines which of the numbers in `f` are greater than zero, then returns indices for which this is true, and then returns the values at those indices and assigns them as a vector to “`g`”.
- E. `>> x = zeros(1,6)+0.5; >> y = 0.5.*ones(1,length(x)); >> z = x + y;` A vector of six zeros is generated and then increased by 0.5 and assigned to “`x`.” A vector of ones the length of `x` is generated, increased by 0.5, and assigned to “`y`.” Finally, the vector resulting from the sum of `x` and `y` is assigned to “`z`.”
- F. `>> a = [1:6]; >> b = a([end:-1:1]);` A vector of length six including the integers `1,2,...6` is assigned to “`a`.” A vector “`b`” is created which begins with the final entry in `a` and then includes every index in `a` decrementing by -1 until index 1 is reached. I.e., `b` is a reversed.
3. A.
`function result = diceRoll()`
`result = randi(6,1,1);`
`end`
- B.
`>> y = [1,2,3,4,5,6]; >> reshape(y, [2,3]) ;`
- C.
`>> minNum = min(min(z));`
`>> c = mod(find(z == min(min(z))), length(z))+1;`
`>> r = floor((find(z == min(min(z)))/length(z))+1;`

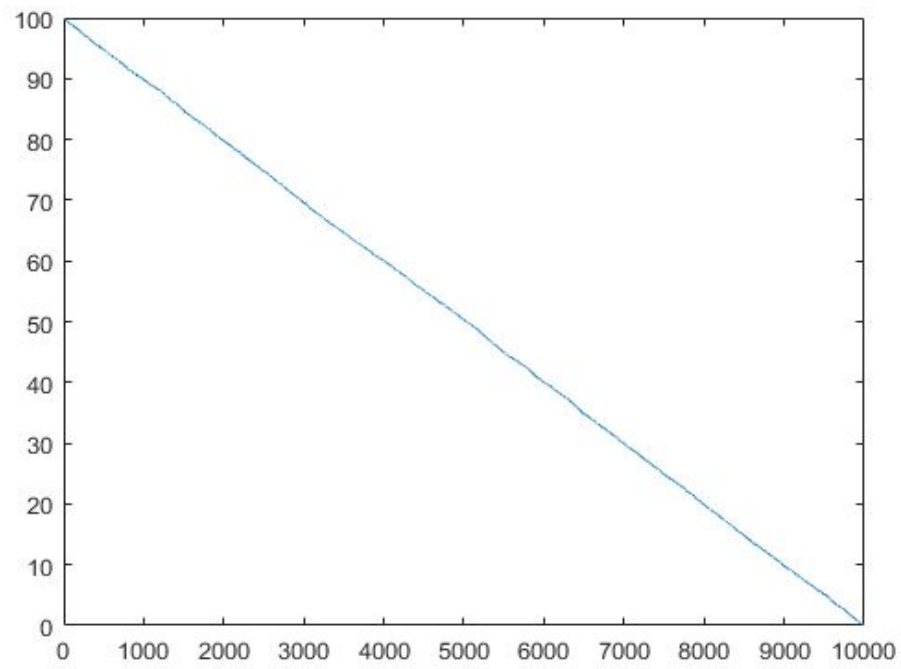
D.

```
>> v = [8, 8, 3, 2, 1, 8, 1, 8];
```

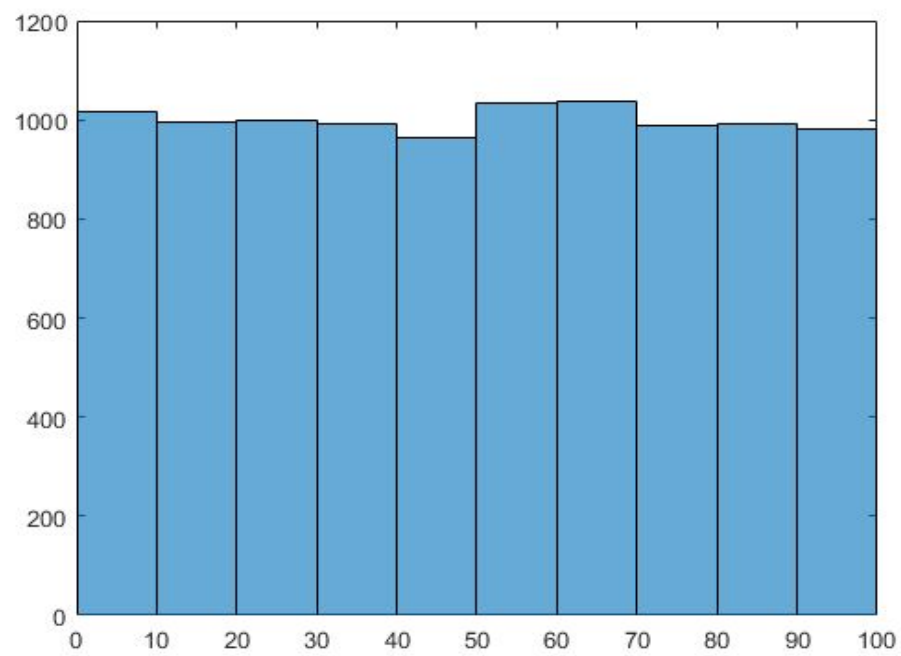
```
>> x = length(find(v==8));
```

4.

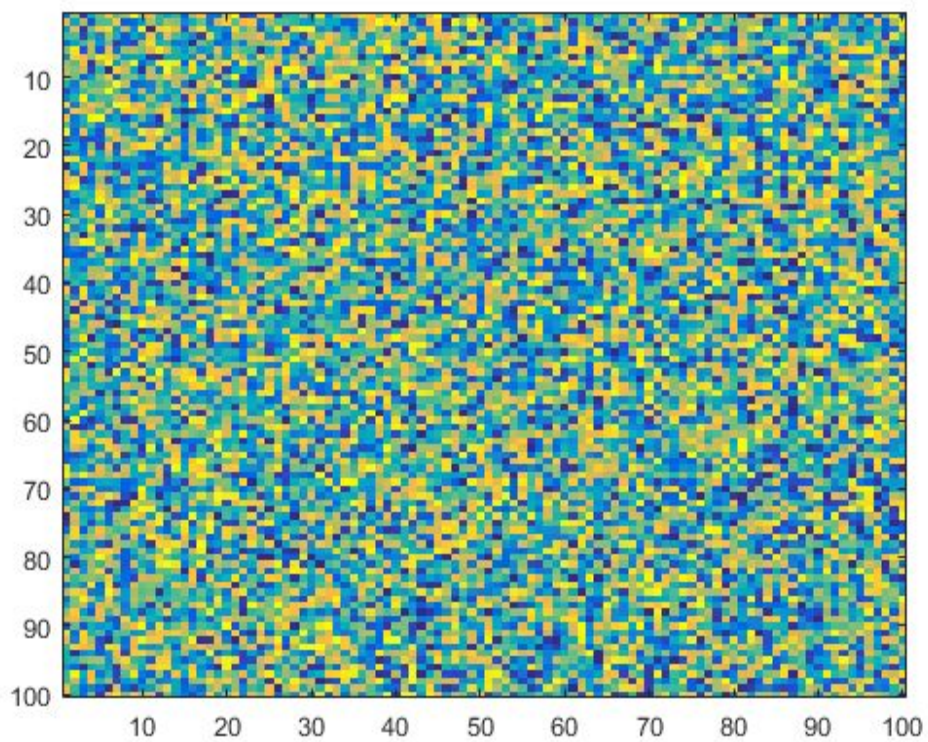
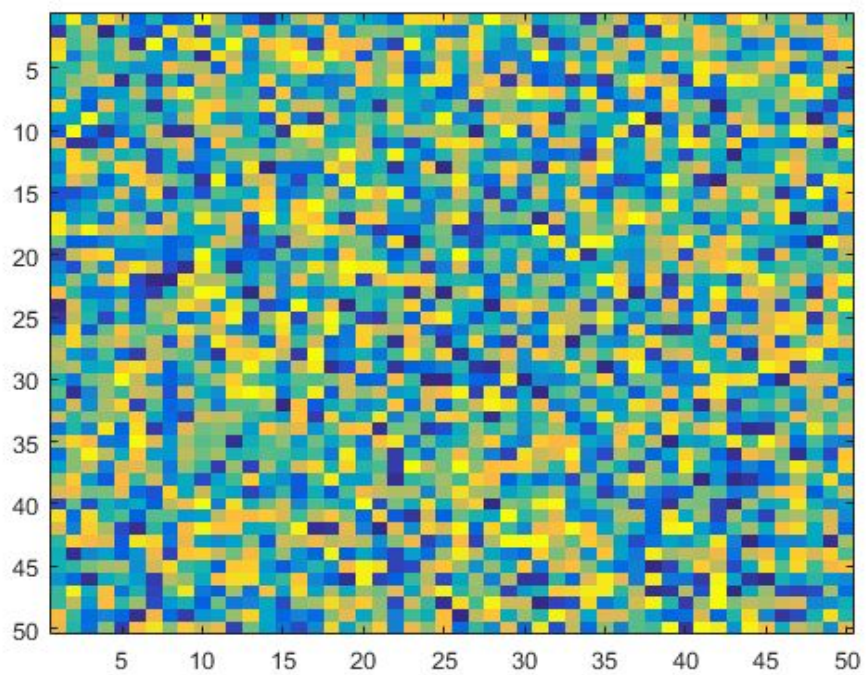
A.



B.

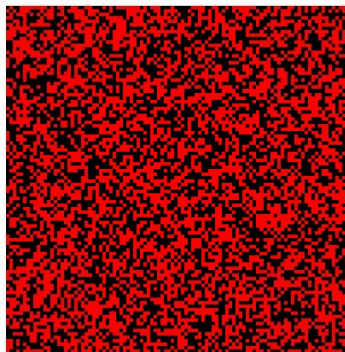


C.



D.

E.



Part II.

