a) I use the os.urandom function to generate the bits.

b)

```
mastermindccr@mastermindccr:~/Desktop/CE/sts-2.1.2$ ./assess 8388608
            G E N E R A T O R    S E L E C T I O N
            ----------------------------------

      [0] Input File                [1] Linear Congruential
      [2] Quadratic Congruential I  [3] Quadratic Congruential II
      [4] Cubic Congruential        [5] XOR
      [6] Modular Exponentiation    [7] Blum-Blum-Shub
      [8] Micali-Schnorr            [9] G Using SHA-1

     Enter Choice: 0


                User Prescribed Input File: ../random.bin

                S T A T I S T I C A L   T E S T S
                ---------------------------------

      [01] Frequency                       [02] Block Frequency
      [03] Cumulative Sums                 [04] Runs
      [05] Longest Run of Ones             [06] Rank
      [07] Discrete Fourier Transform      [08] Nonperiodic Template Matchings
      [09] Overlapping Template Matchings  [10] Universal Statistical
      [11] Approximate Entropy             [12] Random Excursions
      [13] Random Excursions Variant       [14] Serial
      [15] Linear Complexity

           INSTRUCTIONS
              Enter 0 if you DO NOT want to apply all of the
              statistical tests to each sequence and 1 if you DO.

     Enter Choice: 1

          P a r a m e t e r   A d j u s t m e n t s
          -----------------------------------------
      [1] Block Frequency Test - block length(M):         128
      [2] NonOverlapping Template Test - block length(m): 9
      [3] Overlapping Template Test - block length(m):    9
      [4] Approximate Entropy Test - block length(m):     10
      [5] Serial Test - block length(m):                  16
      [6] Linear Complexity Test - block length(M):       500

     Select Test (0 to continue): 1

     Enter Block Frequency Test block length: 65536

          P a r a m e t e r   A d j u s t m e n t s
          -----------------------------------------
      [1] Block Frequency Test - block length(M):         65536
      [2] NonOverlapping Template Test - block length(m): 9
      [3] Overlapping Template Test - block length(m):    9
      [4] Approximate Entropy Test - block length(m):     10
      [5] Serial Test - block length(m):                  16
      [6] Linear Complexity Test - block length(M):       500
```

```
Select Test (0 to continue): 0

How many bitstreams? 1

Input File Format:
 [0] ASCII - A sequence of ASCII 0's and 1's
 [1] Binary - Each byte in data file contains 8 bits of data

Select input mode:  1

  Statistical Testing In Progress.........

  Statistical Testing Complete!!!!!!!!!!!!
```

```
------------------------------------------------------------------------------
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES
------------------------------------------------------------------------------
   generator is <../random.bin>
------------------------------------------------------------------------------
 C1  C2  C3  C4  C5  C6  C7  C8  C9 C10  P-VALUE  PROPORTION  STATISTICAL TEST
------------------------------------------------------------------------------
  0   0   0   0   0   0   0   0   0   1     ----       1/1       Frequency
  0   0   0   0   0   0   0   0   0   1     ----       1/1       BlockFrequency
  0   0   0   0   0   0   0   0   1   0     ----       1/1       CumulativeSums
  0   0   0   0   0   0   0   0   0   1     ----       1/1       CumulativeSums
  0   0   0   0   0   0   0   0   0   1     ----       1/1       Runs
  0   0   0   0   0   0   1   0   0   0     ----       1/1       LongestRun
  0   0   0   1   0   0   0   0   0   0     ----       1/1       Rank
  1   0   0   0   0   0   0   0   0   0     ----       1/1       FFT
  0   0   0   0   1   0   0   0   0   0     ----       1/1       NonOverlappingTemplate
  0   0   0   0   0   0   0   0   0   1     ----       1/1       NonOverlappingTemplate
  0   0   0   0   0   0   1   0   0   0     ----       1/1       NonOverlappingTemplate
  0   0   0   0   0   0   0   0   1   0     ----       1/1       NonOverlappingTemplate
  0   0   1   0   0   0   0   0   0   0     ----       1/1       NonOverlappingTemplate
  0   0   0   0   0   0   0   0   0   1     ----       1/1       NonOverlappingTemplate
  0   0   0   0   1   0   0   0   0   0     ----       1/1       NonOverlappingTemplate
  0   0   0   0   1   0   0   0   0   0     ----       1/1       NonOverlappingTemplate
  0   0   1   0   0   0   0   0   0   0     ----       1/1       NonOverlappingTemplate
  0   1   0   0   0   0   0   0   0   0     ----       1/1       NonOverlappingTemplate
  0   0   0   0   0   0   1   0   0   0     ----       1/1       NonOverlappingTemplate
  0   1   0   0   0   0   0   0   0   0     ----       1/1       NonOverlappingTemplate
  0   0   0   0   0   0   0   0   0   1     ----       1/1       NonOverlappingTemplate
  0   0   0   0   0   0   1   0   0   0     ----       1/1       NonOverlappingTemplate
```

30	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
31	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
32	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
33	0	0	0	0	0	0	0	0	0	1	----	1/1	NonOverlappingTemplate
34	0	0	0	0	0	0	0	0	0	1	----	1/1	NonOverlappingTemplate
35	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
36	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
37	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
38	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
39	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
40	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
41	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTem匚late
42	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
43	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
44	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
45	0	0	0	0	0	0	0	0	0	1	----	1/1	NonOverlappingTemplate
46	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
47	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
48	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
49	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
50	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
51	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
52	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
53	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
54	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
55	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
56	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
57	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
58	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
59	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
60	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
61	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
62	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
63	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
64	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
65	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
66	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
67	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
68	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
69	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
70	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
71	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
72	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
73	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
74	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
75	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
76	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
77	0	0	0	0	0	0	0	0	0	1	----	1/1	NonOverlappingTemplate
78	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
79	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
80	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
81	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
82	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
83	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
84	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
85	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
86	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
87	1	0	0	0	0	0	0	0	0	0	----	0/1	NonOverlappingTemplate

88	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
89	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
90	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
91	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
92	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
93	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
94	0	0	0	0	0	0	0	0	0	1	----	1/1	NonOverlappingTemplate
95	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
96	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
97	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
98	0	0	0	0	0	0	0	0	0	1	----	1/1	NonOverlappingTemplate
99	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
100	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
101	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
102	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
103	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
104	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
105	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
106	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
107	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
108	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
109	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
110	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
111	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
112	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
113	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
114	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
115	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
116	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate

117	0	0	0	0	0	0	0	0	0	1	----	1/1	NonOverlappingTemplate
118	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
119	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
120	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
121	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
122	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
123	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
124	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
125	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
126	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
127	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
128	0	0	0	0	0	1	0	0	0	0	----	1/1	NonOverlappingTemplate
129	1	0	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
130	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
131	0	0	0	0	0	0	1	0	0	0	----	1/1	NonOverlappingTemplate
132	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
133	0	0	0	0	1	0	0	0	0	0	----	1/1	NonOverlappingTemplate
134	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
135	0	0	1	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
136	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
137	0	1	0	0	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
138	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
139	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
140	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
141	0	0	0	0	0	0	0	1	0	0	----	1/1	NonOverlappingTemplate
142	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate
143	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
144	0	0	0	1	0	0	0	0	0	0	----	1/1	NonOverlappingTemplate
145	0	0	0	0	0	0	0	0	1	0	----	1/1	NonOverlappingTemplate

```
 0  0  0  0  0  0  0  1  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  0  0  1  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  0  0  0  1    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  0  1  0  0    ----    1/1    NonOverlappingTemplate
 0  1  0  0  0  0  0  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  1  0  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  1  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  1  0  0  0  0  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  1  0  0  0  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  0  0  1  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  1  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  1  0  0  0  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  0  0  0  1    ----    1/1    NonOverlappingTemplate
 0  0  1  0  0  0  0  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  1  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  0  0  0  1  0    ----    1/1    NonOverlappingTemplate
 1  0  0  0  0  0  0  0  0  0    ----    0/1    NonOverlappingTemplate
 0  1  0  0  0  0  0  0  0  0    ----    1/1    NonOverlappingTemplate
 0  0  0  0  0  1  0  0  0  0    ----    1/1    OverlappingTemplate
 0  0  0  0  0  1  0  0  0  0    ----    1/1    Universal
 0  0  0  0  1  0  0  0  0  0    ----    1/1    ApproximateEntropy
 0  0  0  1  0  0  0  0  0  0    ----    1/1    RandomExcursions
 0  0  0  0  1  0  0  0  0  0    ----    1/1    RandomExcursions
 0  0  0  0  0  0  0  0  0  1    ----    1/1    RandomExcursions
 0  0  0  0  0  0  0  0  1  0    ----    1/1    RandomExcursions
 0  0  0  0  0  0  0  0  0  1    ----    1/1    RandomExcursions
 0  0  0  0  0  1  0  0  0  0    ----    1/1    RandomExcursions
 0  0  0  0  0  0  0  1  0  0    ----    1/1    RandomExcursions
 1  0  0  0  0  0  0  0  0  0    ----    0/1    RandomExcursions
```

```
 0  0  0  0  0  0  1  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  0  1  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  1  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  1  0  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  1  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  1  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  1  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  1  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  1  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  0  1  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  0  0  0  0  1  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  0  0  0  1  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  1  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  1  0  0  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 1  0  0  0  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 1  0  0  0  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  1  0  0  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  1  0  0  0  0  0  0  0  0    ----    1/1    RandomExcursionsVariant
 0  0  0  0  0  1  0  0  0  0    ----    1/1    Serial
 0  0  0  0  1  0  0  0  0  0    ----    1/1    Serial
 0  0  0  1  0  0  0  0  0  0    ----    1/1    LinearComplexity


- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
The minimum pass rate for each statistical test with the exception of the
random excursion (variant) test is approximately = 0 for a
sample size = 1 binary sequences.

The minimum pass rate for the random excursion (variant) test
```

```
204  is approximately = 0 for a sample size = 1 binary sequences.
205
206  For further guidelines construct a probability table using the MAPLE program
207  provided in the addendum section of the documentation.
208  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
209
```

Analysis:

- Frequency test: compute the number of 0s and 1s and calculate the difference between them to calculate the p-value. The test fails when the frequency of generating them differs too much.
- Block Frequency test: partition the bits into blocks with n bits inside each block. Calculate the frequency of 1s in each block first and then perform chi-square statistics. The test fails when the frequency in each block differs too much, which means it does not distribute equally.
- Runs test: Based on the frequency test and further test the oscillation of 0s and 1s. The test fails when the oscillation between consecutive bits is too fast or slow.
- Longest run of 1s test: test whether the longest run of 1s in a block is more than expected. It uses the chi-square test to calculate the p-value from the probability of generating such a long run. The test fails when the observed longest run of 1s is much more than expected.
- Rank test: check for linear dependence among fixed length substrings of the original sequence by calculating the rank of disjoint sub-matrices of the entire sequence. The chi-square test fails when the observed rank does not match the expected number under an assumption of randomness.
- Discrete Fourier Transform test: test whether there's repetitive patterns near to each other. The test fails when it detects the number of peaks exceeding the 95% threshold is significantly different than 5%.
- Non-overlapping(periodic) Template Matchings test: test whether the sequence has too much pre-specified sequence. If the pattern is not found, the search window slides for one bit. The test fails if the observed number of the specified sequence is much more than expected.
- Overlapping Template Matching test: almost the same as the non-

overlapping test, the only difference is that when the pattern is found, the search window slides for one bit.

- Universal Statistical test: the test detects whether the sequence can be significantly compressed without losing information. The test fails when it can be significantly compressed, as it shows that there are several same consecutive subsequences in the original sequence.
- Linear Complexity test: the test calculate how long it needs to characterize the sequence with a LFSR. The test fails if the LFSR is too short, which implies the lack of randomness.
- Serial test: the test enumerates and calculates the number of all the subsequences within a certain m bit (which as 2^m possibilities). The test fails if the calculated frequency differs from the one calculated from a true random sequence.
- Approximate Entropy test: use the same way as in the serial test, the only difference is that it only compares the neighboring blocks and see whether the match the expected result for a random sequence.
- Cumulative Sums test: the test sums the sequence from the beginning and see that whether the largest and the smallest number is too large or small (exceed the expected range of true randomness). The test fails if there are too many 0s or 1s in the first k bits.
- Random Excursions test: The focus of this test is the number of cycles having exactly K visits in a cumulative sum random walk. The test will be performed in several times to check each state. The test fails if a particular state is entered too frequently.
- Random Excursions Variant test: Almost the same as in random excursions test, the difference is that the test only focuses on the total number of times that a particular state is visited in a cumulative sum random walk.

P.S. You can simply use “python3 RNG.py” to run my code and generate random.bin