| # | | Focus | Purpose | Input size recommendation |
|---|---|---|---|---|
| **1** | The Frequency (Monobit) Test | test is the proportion of zeroes and ones for the entire sequence | determine whether the number of ones and zeros in a sequence are approximately the same as would be expected for a truly random sequence | $n \geq 100$ |
| **2** | Frequency Test within a Block | proportion of ones within M-bit blocks | determine whether the frequency of ones in an M-bit block is approximately M/2, as would be expected under an assumption of randomness | $n \geq 100$ |
| **3** | The Runs Test | total number of runs in the sequence, where a run is an uninterrupted sequence of identical bits | determine whether the number of runs of ones and zeros of various lengths is as expected for a random sequence (determines whether the oscillation between such zeros and ones is too fast or too slow) | $n \geq 100$ |
| **4** | Tests for the Longest-Run-of-Ones in a Block | longest run of ones within M-bit blocks | determine whether the length of the longest run of ones within the tested sequence is consistent with the length of the longest run of ones that would be expected in a random sequence | $n \geq 128$ ; 6,272 ; 750,000 bits |
| **5** | The Binary Matrix Rank Test | rank of disjoint sub-matrices of the entire sequence | check for linear dependence among fixed length substrings of the original sequence | minimum number of bits to be tested must be such that $n \geq 38MQ$. For M = Q = 32, each sequence to be tested should consist of a minimum of 38,912 bits |
| **6** | The Discrete Fourier Transform (Spectral) Test | peak heights in the Discrete Fourier Transform of the sequence | detect periodic features (i.e., repetitive patterns that are near each other) in the tested sequence that would indicate a deviation from the assumption of randomness, detect whether the number of peaks exceeding the 95 % threshold is significantly different than 5 % | $n \geq 1{,}000$ |
| **7** | The Non-overlapping Template Matching Test | focus of this test is the number of occurrences of pre-specified target strings | detect generators that produce too many occurrences of a given non-periodic (aperiodic) pattern (related to Test #8) | be sure that $M > 0.01 \cdot n$ and $N = floor(n/M)$ ; for more see Section 2.7.7 |
| **8** | The Overlapping Template Matching Test | focus of the Overlapping Template Matching test is the number of occurrences of pre-specified target strings | detect generators that produce too many occurrences of a given non-periodic (aperiodic) pattern (related to Test #7) | $n \geq 1{,}000{,}000$ ; for more see Section 2.8.7 |
| **9** | Maurer's "Universal Statistical" Test | number of bits between matching patterns (a measure that is related to the length of a compressed sequence) | detect whether or not the sequence can be significantly compressed without loss of information. A significantly compressible sequence is considered to be non-random | $n \geq 387{,}840$; $\geq 904{,}960$ ; $\geq 2{,}068{,}480$; $\geq 4{,}654{,}080$; $\geq 10{,}342{,}400$; $\geq 22{,}753{,}280$; $\geq 49{,}643{,}520$; $\geq 107{,}560{,}960$; $\geq 231{,}669{,}760$; $\geq 496{,}435{,}200$; $\geq 1{,}059{,}061{,}760$ ; for more see Section 2.9.7 |
| **10** | The Linear Complexity Test | the length of a linear feedback shift register (LFSR) | determine whether or not the sequence is complex enough to be considered random. Random sequences are characterized by longer LFSRs. An LFSR that is too short implies non-randomness | $n \geq 1{,}000{,}000$ ; more details in Section 2.10.7 |
| **11** | The Serial Test | frequency of all possible overlapping *m*-bit patterns across the entire sequence | determine whether the number of occurrences of the 2^m *m*-bit overlapping patterns is approximately the same as would be expected for a random sequence | Choose $m < floor(log2(n)) - 2$ ; example n = 1,000,000 |
| **12** | The Approximate Entropy Test | frequency of all possible overlapping *m*-bit patterns across the entire sequence | compare the frequency of overlapping blocks of two consecutive/adjacent lengths (m and m+1) against the expected result for a random sequence | Choose $m < floor(log2(n)) - 5$ ; example n = 100 |
| **13** | The Cumulative Sums (Cusums) Test | maximal excursion (from zero) of the random walk defined by the cumulative sum of adjusted (-1, +1) digits in the sequence | determine whether the cumulative sum of the partial sequences occurring in the tested sequence is too large or too small relative to the expected behavior of that cumulative sum for random sequences. This cumulative sum may be considered as a random walk. For a random sequence, the excursions of the random walk should be near zero. For certain types of non-random sequences, the excursions of this random walk from zero will be large | $n \geq 100$ |
| **14** | The Random Excursions Test | number of cycles having exactly K visits in a *cumulative sum* random walk. The cumulative sum random walk is derived from partial sums after the *(0,1)* sequence is transferred to the appropriate *(-1, +1)* sequence. A cycle of a random walk consists of a sequence of steps of unit length taken at random that begin at and return to the origin | determine if the number of visits to a particular state within a cycle deviates from what one would expect for a random sequence. This test is actually a series of eight tests (and conclusions), one test and conclusion for each of the states: -4, -3, -2, -1 and +1, +2, +3, +4 | each sequence to be tested consist of a minimum $n \geq 1{,}000{,}000$ |
| **15** | The Random Excursions Variant Test | total number of times that a particular state is visited (i.e., occurs) in a cumulative sum random walk | detect deviations from the expected number of visits to various states in the random walk. This test is actually a series of eighteen tests (and conclusions), one test and conclusion for each of the states: -9, -8, …, -1 and +1, +2, …, +9 | each sequence to be tested consist of a minimum $n \geq 1{,}000{,}000$ |