

Statistical Tests for Random Number Generators

Venkata Koonaparaju

Introduction

- Random Number Generators

Ex: Flipping a unbiased coin, Radio active decay

- Applications

Ex: One time pad, Generation of Primes, Secret Key

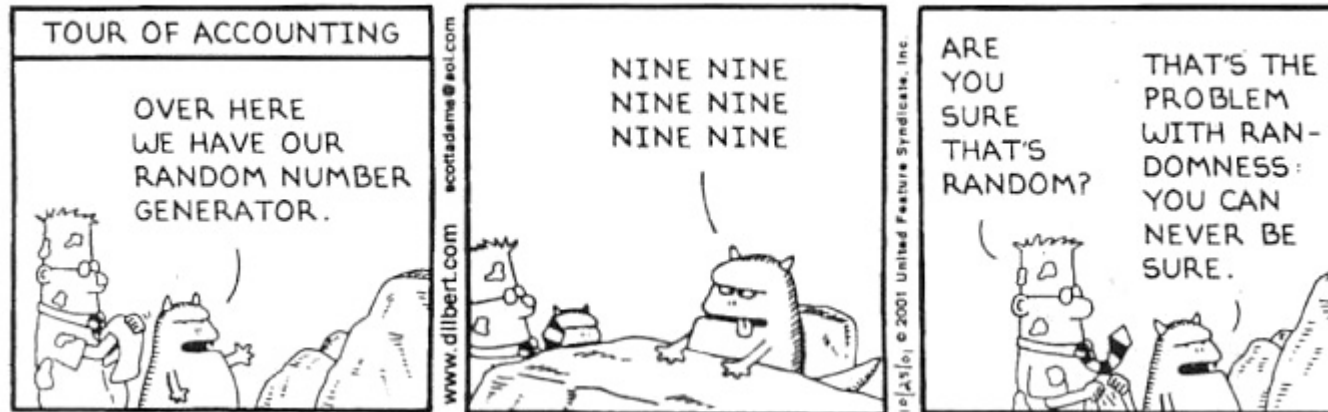
Types of Generators

- True Random Number Generators
- Pseudo Random Number Generators
- Hybrid Random Number Generators

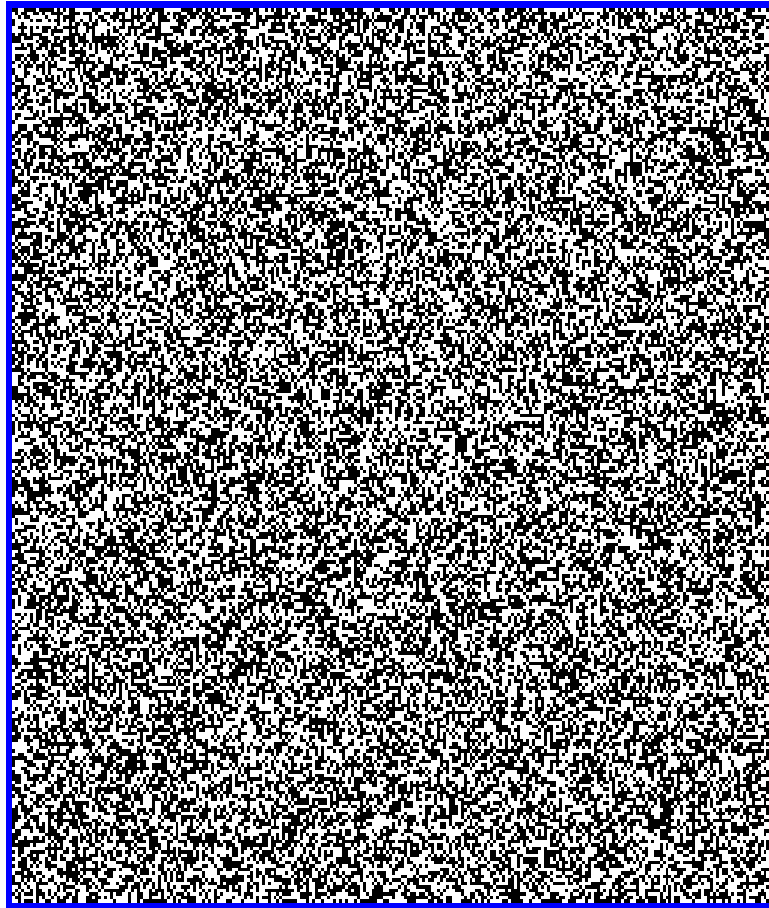
Criteria	TRNG	PRNG
Source	Random Processes like Mouse Movements, IO buffers	Seed
Time required	Requires more time as post processing is required	Faster than TRNG
Secure	Depends on the number of sources used for generation	Depends on secrecy of the seed.

Testing

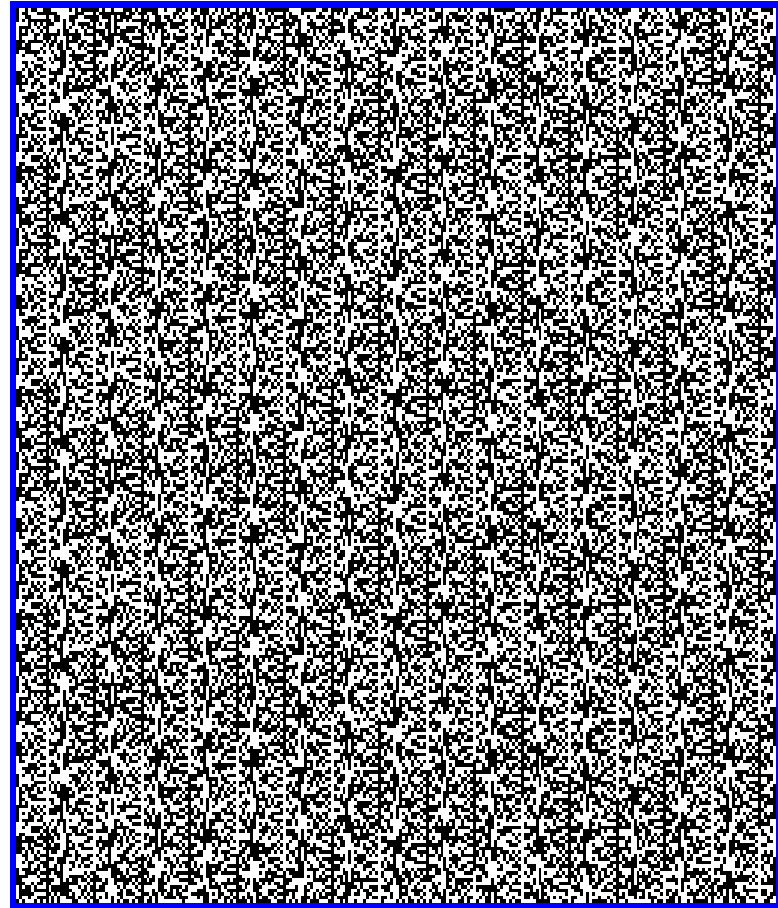
DILBERT By SCOTT ADAMS



Visual Inspection



RANDOM.ORG



PHP rand() on Microsoft Windows

Statistical Tests

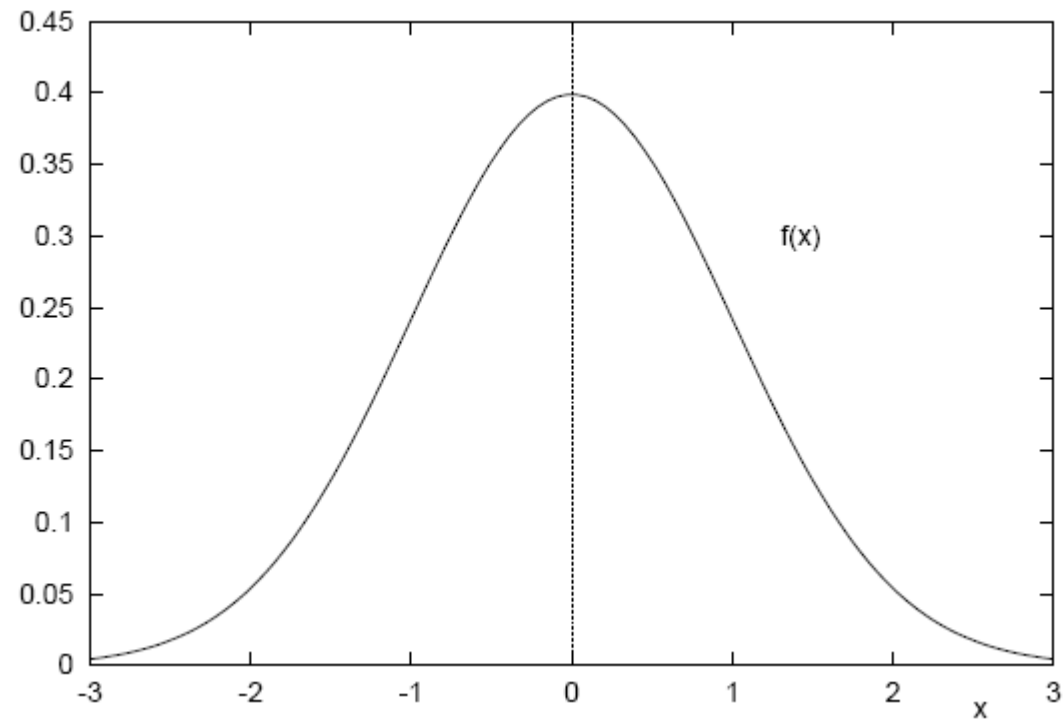
- Randomness is a probabilistic property.

TRUE SITUATION	CONCLUSION	
	Accept H_0	Accept H_a (reject H_0)
Data is random (H_0 is true)	No error	Type I error
Data is not random (H_a is true)	Type II error	No error

Statistical Tests cont..

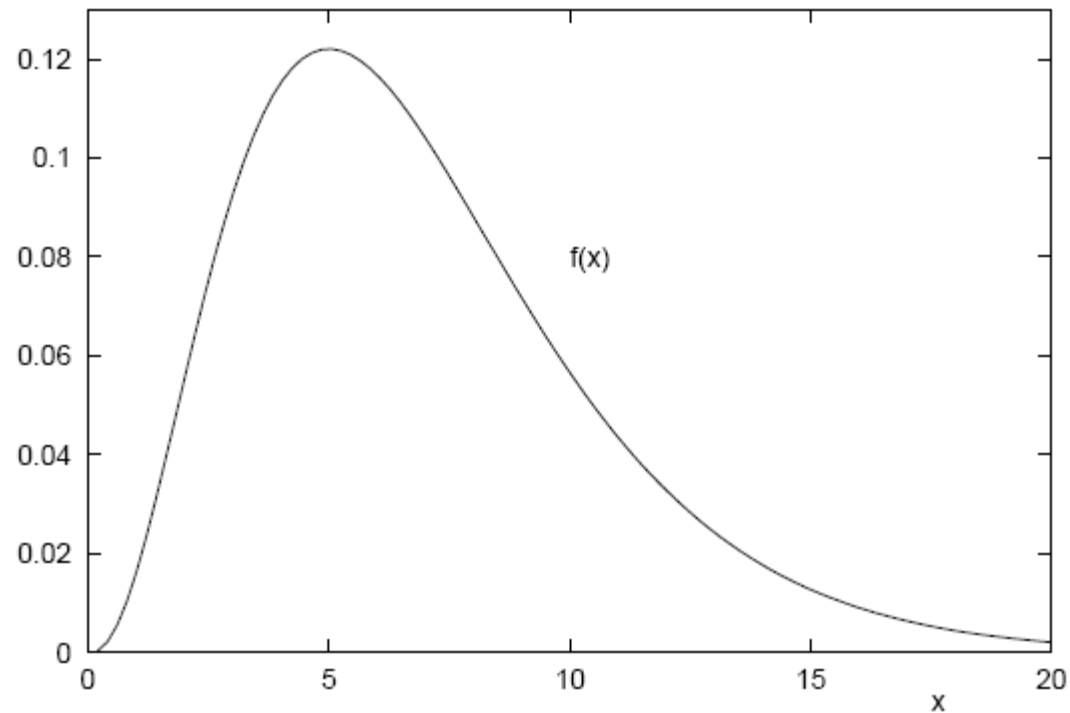
- Diehard
- Crypt-x
- National Institute of Standards and Technology (NIST)
- ENT

Normal Distribution



The Normal Distribution $N(0,1)$

The Chi Square Distribution



The Chi – Square Distribution

Frequency Monobit Test

- Compute the test statistic $sobs = |s|/\sqrt{n}$
- Compute $P\text{-value} = \mathbf{erfc}(sobs/\sqrt{2})$, where \mathbf{erfc} is the complementary error function
- if the P-value were small (< 0.01), then this would be caused by S_n or $sobs$ being large.

Frequency Monobit Test

(input) $\varepsilon = 11001001000011111101101010100010001000010110100011$
00001000110100110001001100011001100010100010111000

(input) $n = 100$

(processing) $S_{100} = -16$

(processing) $s_{obs} = 1.6$

(output) $P\text{-value} = 0.109599$

(conclusion) Since $P\text{-value} \geq 0.01$, accept the sequence as random.

Frequency Test with in a Block

Determine the proportion π_i of ones in each M -bit block using the equation

$$\pi_i = \frac{\sum_{j=1}^M \mathcal{E}_{(i-1)M+j}}{M}, \text{ for } 1 \leq i \leq N.$$

Compute the χ^2 statistic: $\chi^2(\text{obs}) = 4 M \sum_{i=1}^N (\pi_i - 1/2)^2$.

Compute P -value = **igamc** ($N/2, \chi^2(\text{obs})/2$), where **igamc** is the incomplete gamma function

Frequency Test with in a Block

(input) $\varepsilon = 11001001000011111101101010100010001000010110100011$
00001000110100110001001100011001100010100010111000

(input) $n = 100$

(input) $M = 10$

(processing) $N = 10$

(processing) $\chi^2 = 7.2$

(output) $P\text{-value} = 0.706438$

(conclusion) Since $P\text{-value} \geq 0.0$, accept the sequence as random.

Runs Test

Compute the pre-test proportion π of ones in the input sequence: $\pi = \frac{\sum_j \varepsilon_j}{n}$.

Compute the test statistic $V_n(obs) = \sum_{k=1}^{n-1} r(k) + 1$, where $r(k) = 0$ if $\varepsilon_k = \varepsilon_{k+1}$, and $r(k) = 1$ otherwise.

Compute *P-value* = $erfc\left(\frac{|V_n(obs) - 2n\pi(1-\pi)|}{2\sqrt{2n\pi(1-\pi)}}\right)$.

Runs Test

(input) $\varepsilon = 11001001000011111101101010100010001000010110100011$
00001000110100110001001100011001100010100010111000

(input) $n = 100$

(input) $\tau = 0.02$

(processing) $\pi = 0.42$

(processing) $V_n(obs) = 52$

(output) $P\text{-value} = 0.500798$

(conclusion) Since $P\text{-value} \geq 0.01$, accept the sequence as random.

Test for longest run of ones in a block

Tabulate the frequencies v_i of the longest runs of ones in each block into categories, where each cell contains the number of runs of ones of a given length.

For the values of M supported by the test code, the v_i cells will hold the following counts:

v_i	$M = 8$	$M = 128$	$M = 10^4$
v_0	≤ 1	≤ 4	≤ 10
v_1	2	5	11
v_2	3	6	12
v_3	≥ 4	7	13
v_4		8	14
v_5		≥ 9	15
v_6			≥ 16

Test for longest run of ones in a block

$$\text{Compute } \chi^2(\text{obs}) = \sum_{i=0}^K \frac{(v_i - N\pi_i)^2}{N\pi_i}$$

$$\text{Compute } P\text{-value} = \mathbf{igamc} \left(\frac{K}{2}, \frac{\chi^2(\text{obs})}{2} \right).$$

M	K	N
8	3	16
128	5	49
104	6	75

Test for longest run of ones in a block

For the case where $K = 3$ and $M = 8$:

(input) $\varepsilon =$ 1100110000010101011011000100110011100000000001001
00110101010001000100111101011010000000110101111100
1100111001101101100010110010

(processing)	<u>Subblock</u>	<u>Max-Run</u>	<u>Subblock</u>	<u>Max-Run</u>
	11001100	(2)	00010101	(1)
	01101100	(2)	01001100	(2)
	11100000	(3)	00000010	(1)
	01001101	(2)	01010001	(1)
	00010011	(2)	11010110	(2)
	10000000	(1)	11010111	(3)
	11001100	(2)	11100110	(3)
	11011000	(2)	10110010	(2)

(processing) $v_0 = 4; v_1 = 9; v_2 = 3; v_4 = 0; \chi^2 = 4.882457$

(output) $P\text{-value} = 0.180609$

(conclusion) Since the $P\text{-value}$ is ≥ 0.01 , accept the sequence as random.

Binary Matrix Rank

(input) $\varepsilon =$ the first 100,000 binary digits in the expansion of e

(input) $n = 100000, M = Q = 32$ (NOTE: 672 BITS WERE DISCARDED.)

(processing) $N = 97$

(processing) $F_M = 23, F_{M-1} = 60, N - F_M - F_{M-1} = 14$

(processing) $\chi^2 = 1.2619656$

(output) $P\text{-value} = 0.532069$

(conclusion) Since $P\text{-value} \geq 0.01$, accept the sequence as random.

$$\chi^2(\text{obs}) = \frac{(F_M - 0.2888N)^2}{0.2888N} + \frac{(F_{M-1} - 0.5776N)^2}{0.5776N} + \frac{(N - F_M - F_{M-1} - 0.1336N)^2}{0.1336N}.$$

Non Overlapping Template Matching

Partition the sequence into N independent blocks of length M .

For example, if $\varepsilon = 10100100101110010110$, then $n = 20$. If $N = 2$ and $M = 10$, then the two blocks would be 1010010010 and 1110010110.

For the above example, if $m = 3$ and the template $B = 001$, then the examination proceeds as follows:

Bit Positions	Block 1		Block 2	
	Bits	W_1	Bits	W_2
1-3	101	0	111	0
2-4	010	0	110	0
3-5	100	0	100	0
4-6	001 (hit)	Increment to 1	001 (hit)	Increment to 1
5-7	Not examined		Not examined	
6-8	Not examined		Not examined	
7-9	001	Increment to 2	011	1
8-10	010 (hit)	2	110	1

Non Overlapping Template Matching

$$\mu = (M-m+1)/2^m \quad \sigma^2 = M \left(\frac{1}{2^m} - \frac{2m-1}{2^{2m}} \right).$$

$$\text{Compute } \chi^2(\text{obs}) = \sum_{j=1}^N \frac{(W_j - \mu)^2}{\sigma^2}.$$

$$\text{Compute } P\text{-value} = \mathbf{igamc} \left(\frac{N}{2}, \frac{\chi^2(\text{obs})}{2} \right).$$

Overlapping Template Matching

Partition the sequence into N independent blocks of length M .

For example, if $\varepsilon = 10111011110010110100011100101110111110000101101001$, then $n = 50$. If $K = 2$, $M = 10$ and $N = 5$, then the five blocks are 1011101111 , 0010110100 , 0111001011 , 1011111000 , and 0101101001 .

Bit Positions	Bits	No. of occurrences of $B =$ 11
1-2	10	0
2-3	01	0
3-4	11 (hit)	Increment to 1
4-5	11 (hit)	Increment to 2
5-6	10	2
6-7	01	2
7-8	11 (hit)	Increment to 3
8-9	11 (hit)	Increment to 4
9-10	11 (hit)	Increment to 5

$$\text{Compute } \chi^2(\text{obs}) = \sum_{i=0}^5 \frac{(v_i - N\pi_i)^2}{N\pi_i}$$

$$\text{Compute } P\text{-value} = \mathbf{igamc} \left(\frac{5}{2}, \frac{\chi^2(\text{obs})}{2} \right)$$

Universal Test

For example, if $\varepsilon = 01011010011101010111$, then $n = 20$. If $L = 2$ and $Q = 4$, then $K = \lfloor n/L \rfloor - Q = \lfloor 20/2 \rfloor - 4 = 6$. The initialization segment is 01011010; the test segment is 0111010111. The L -bit blocks are shown in the following table:

Block	Type	Contents
1	Initialization Segment	01
2		01
3		10
4		10
5	Test Segment	01
6		11
7		01
8		01
9		01
10		11

Maurer's Universal Test

	Possible L -bit Value			
	00 (saved in T_0)	01 (saved in T_1)	10 (saved in T_2)	11 (saved in T_3)
Initialization	0	2	4	0

For block 5 (the 1st test block): 5 is placed in the “01” row of the table (i.e., T_1), and $sum = \log_2(5-2) = 1.584962501$.

For block 6: 6 is placed in the “11” row of the table (i.e., T_3), and $sum = 1.584962501 + \log_2(6-0) = 1.584962501 + 2.584962501 = 4.169925002$.

Compute the test statistic: $f_n = \frac{1}{K} \sum_{i=Q+1}^{Q+K} \log_2(i - T_j)$

Compute P -value = $erfc\left(\left|\frac{f_n - \text{expectedValue}(L)}{\sqrt{2}\sigma}\right|\right)$

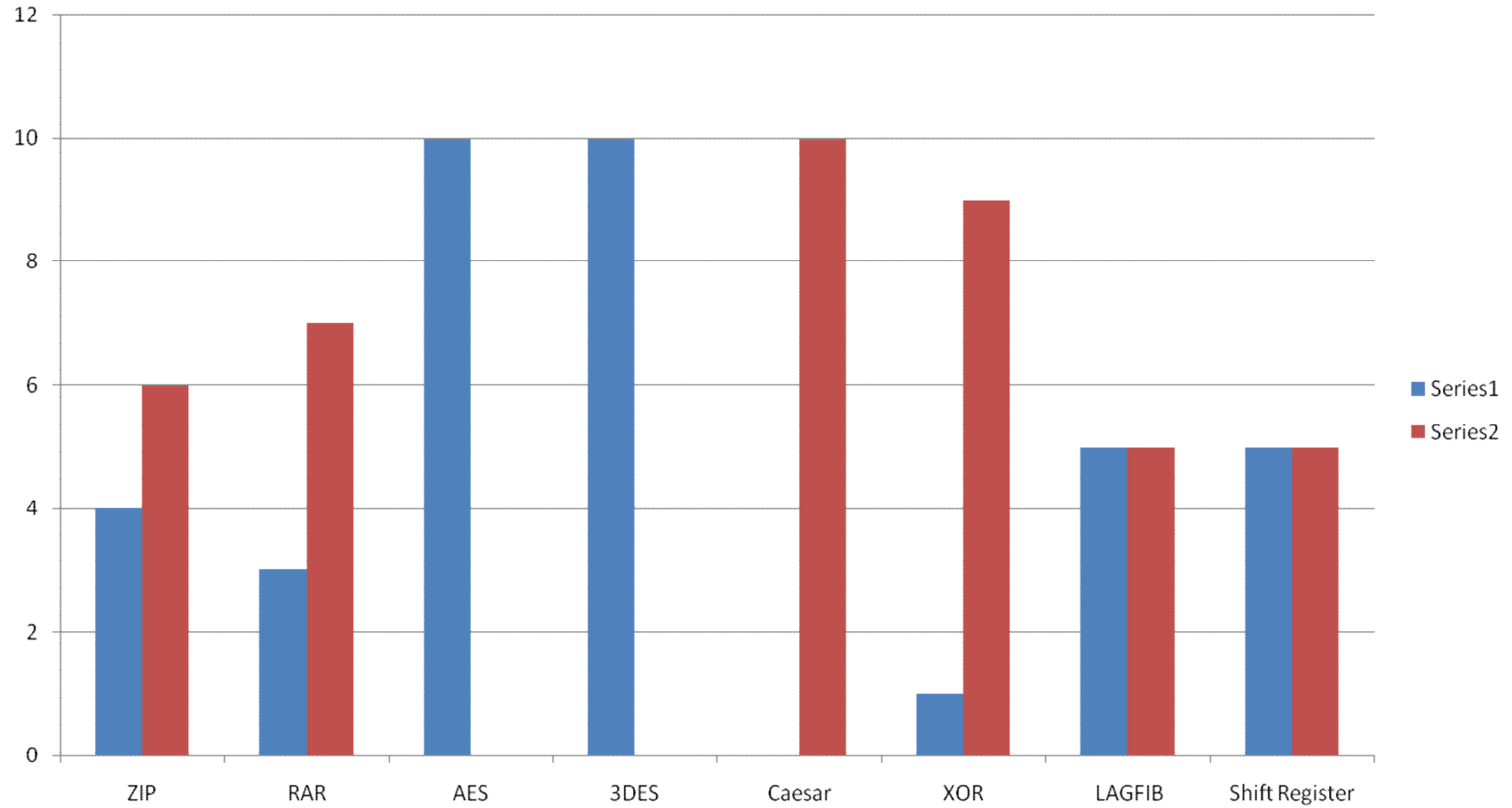
Recommendations

Test Name	Input Size(Minimum)	Other Recommendation
Frequency Monobit	100	
Block Frequency	100	
Runs	100	$M \leq 20$, $M > 0.01n$ and $N < 100$
Longest Run of Ones	128 , 6272, 750,000	$M = 8$, 128, 10^4 respectively
Binary Matrix Rank	38,912	$M=Q=32$
Overlapping Template Matching	10^6	$m = 9$ or 10
Non Overlapping Template Matching	10^6	$m = 9$ or 10
Maurer's Universal Test	387,840	$L = 6$, $Q = 640$

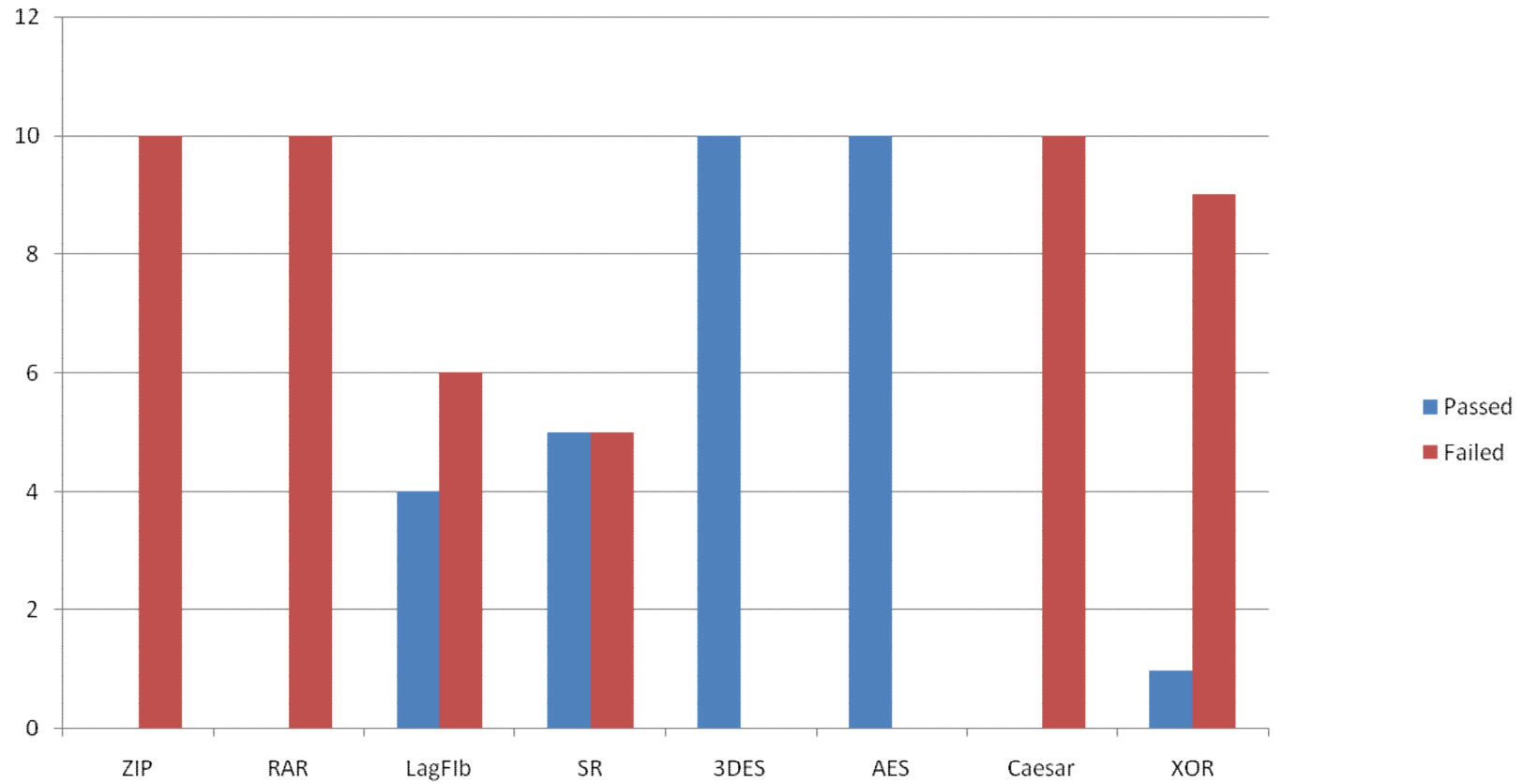
Experiments

- Run the application against compressed, modern ciphers, classic cipher encrypted files and also PSRNG output
- Performance measured in time taken to run the tests.

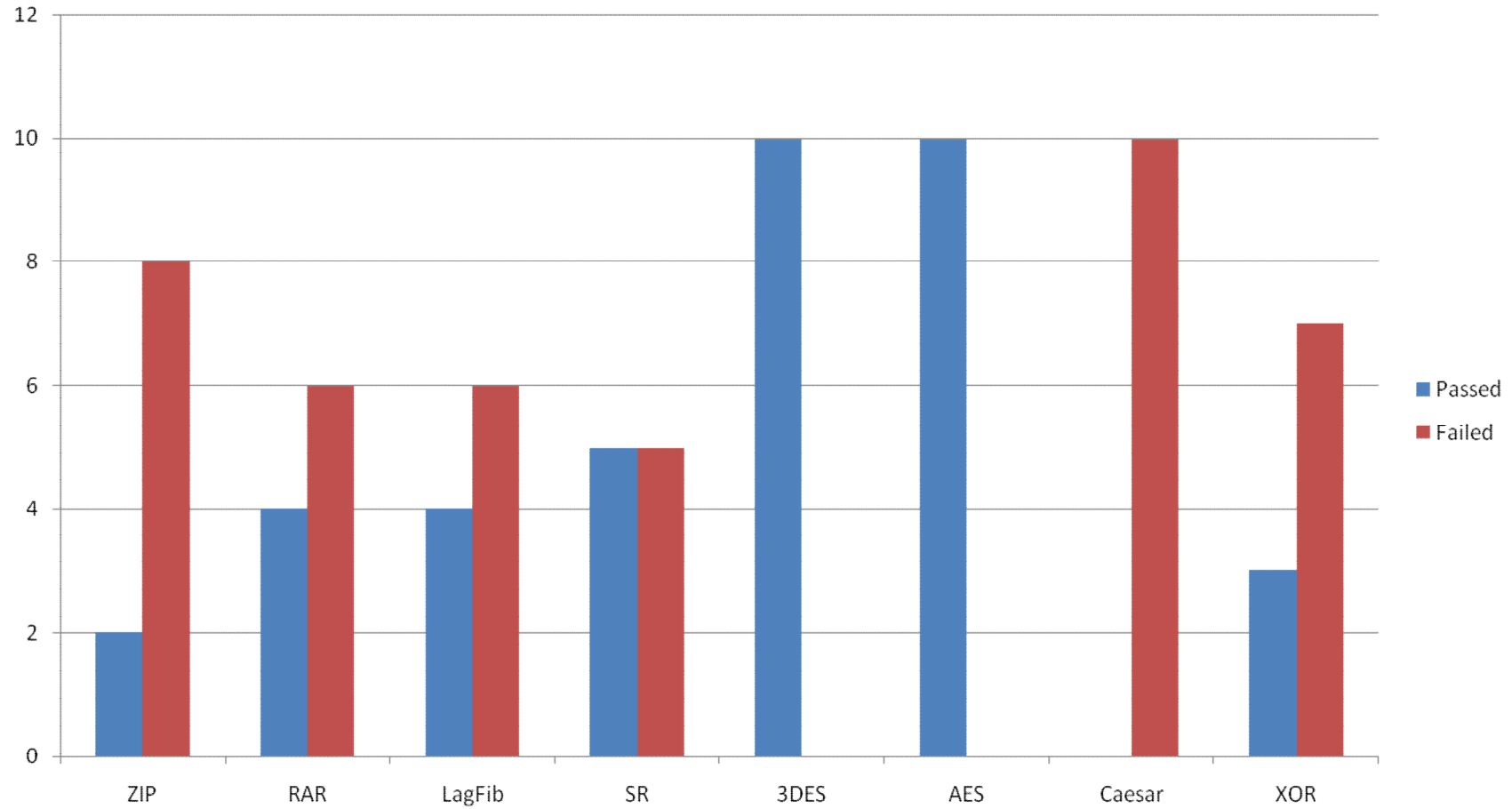
Frequency Monobit



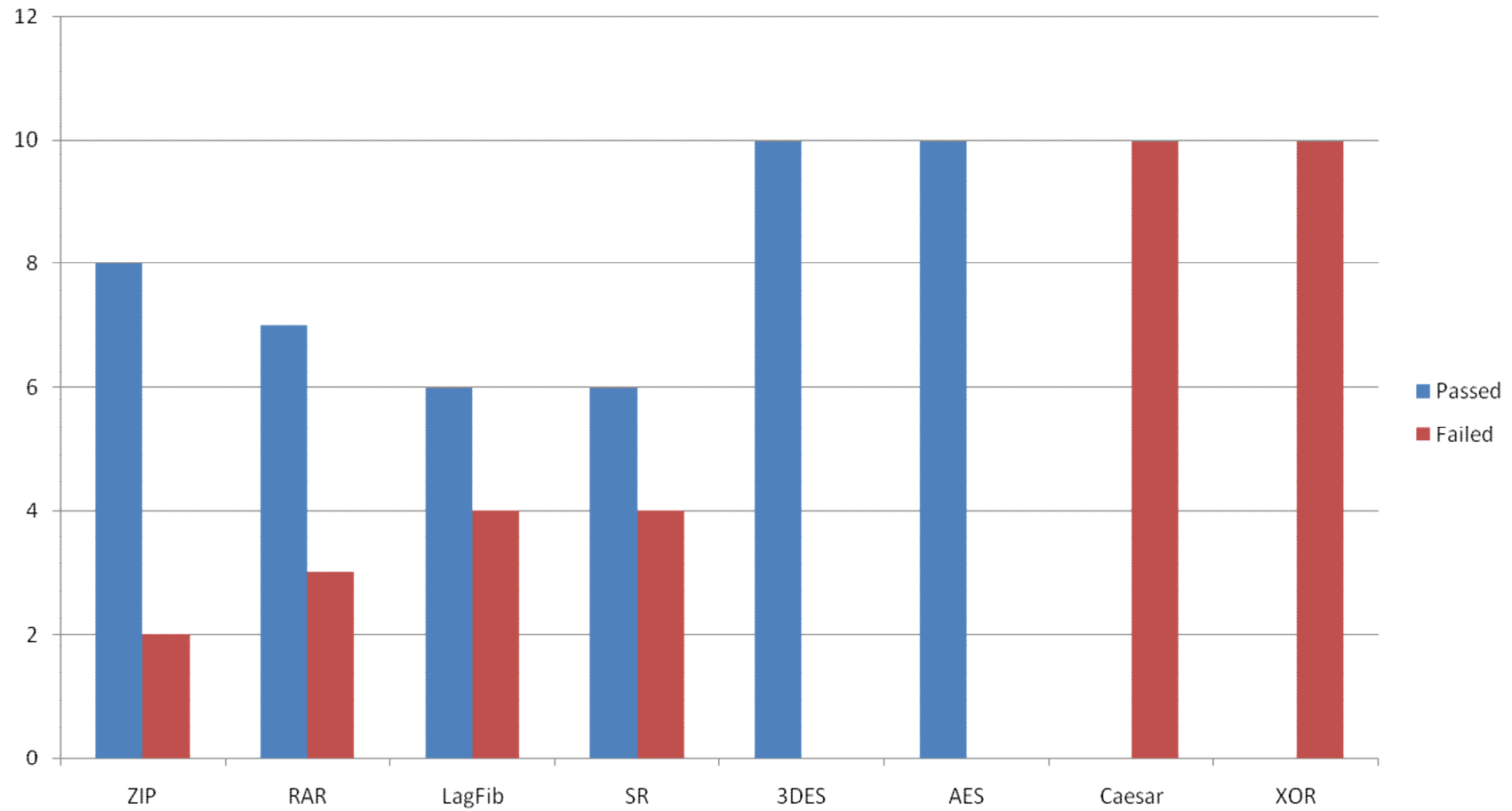
Block Frequency



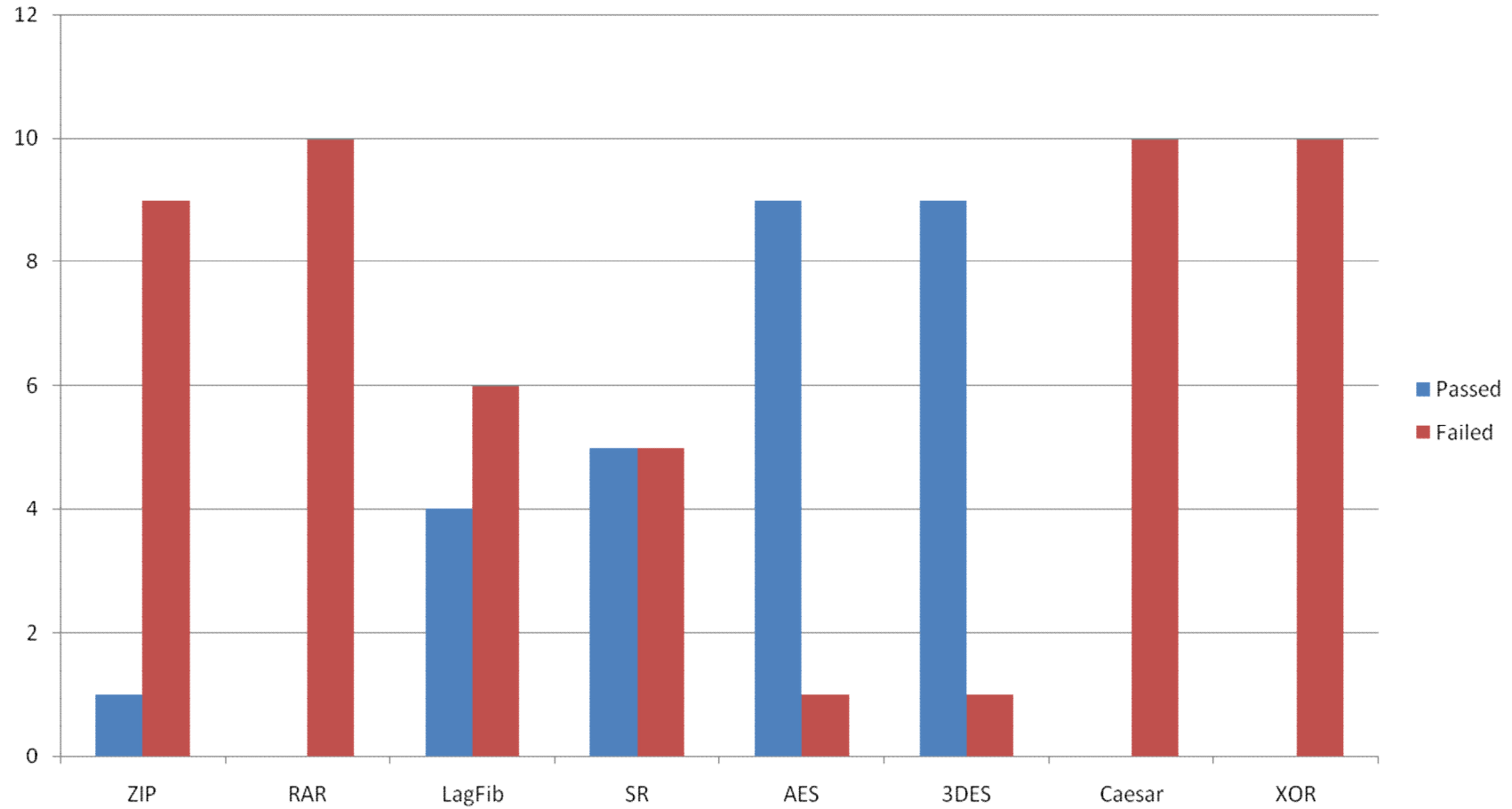
Runs Test



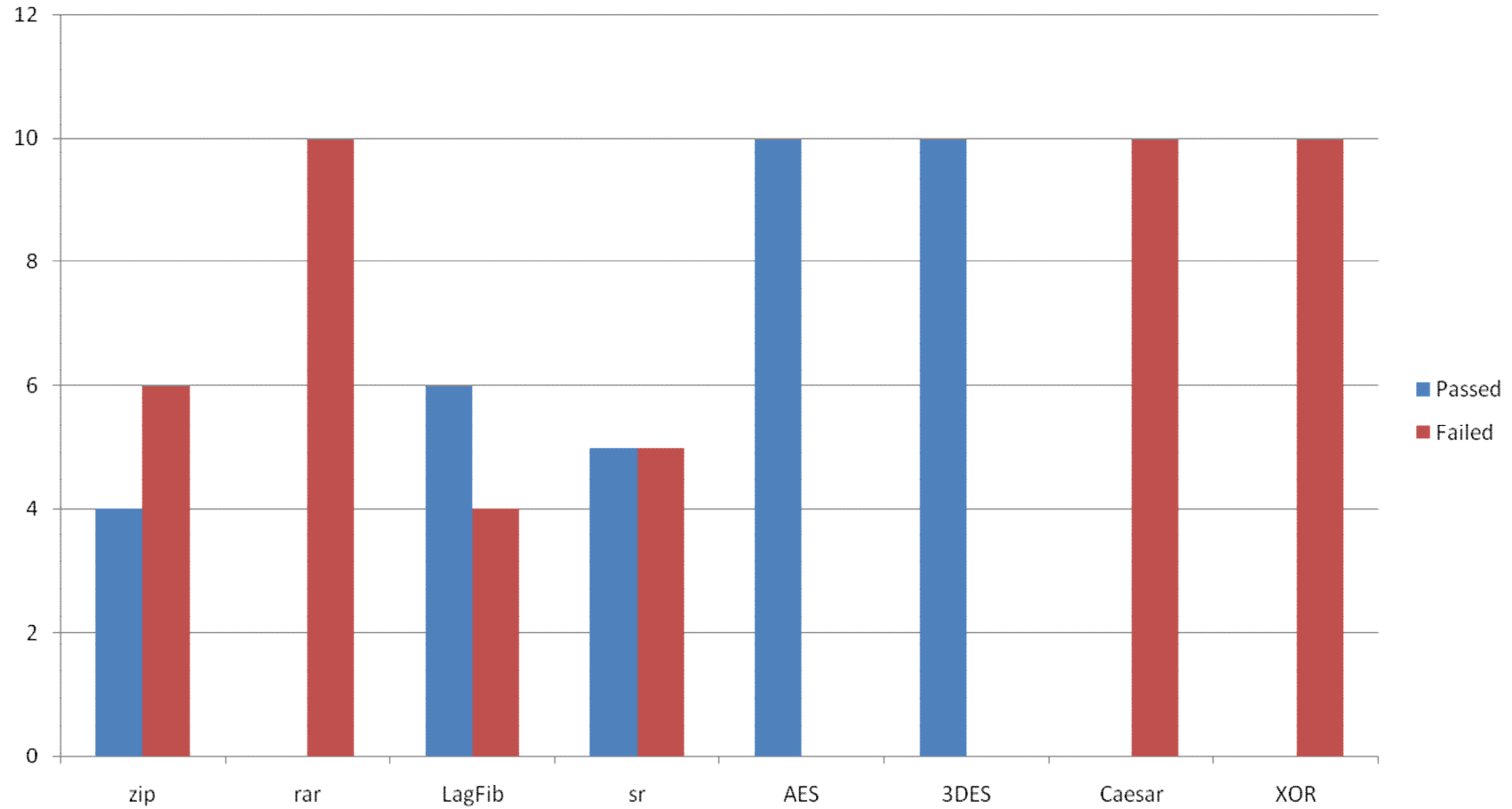
Longest Run of Ones



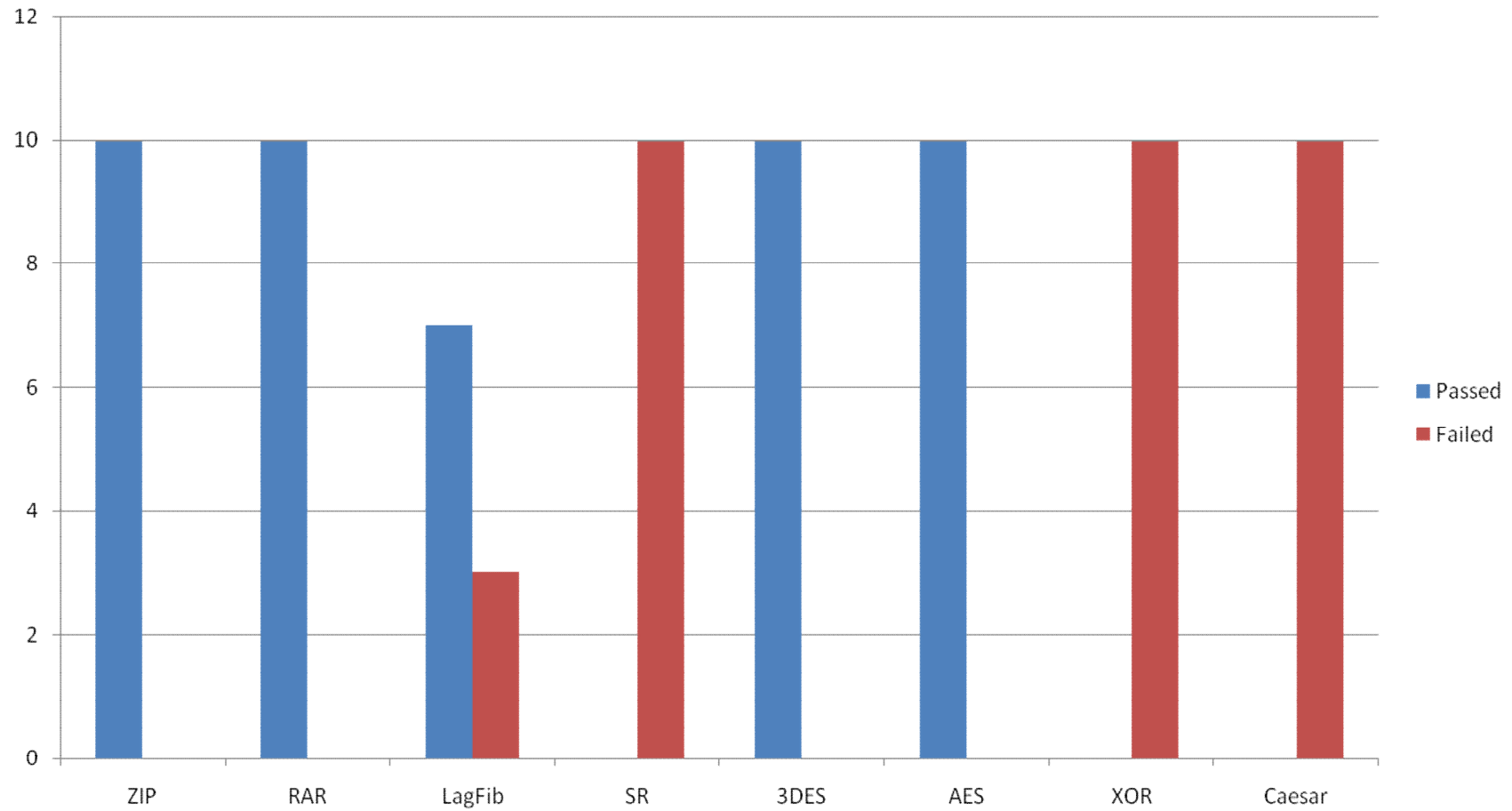
Non Overlapping Template



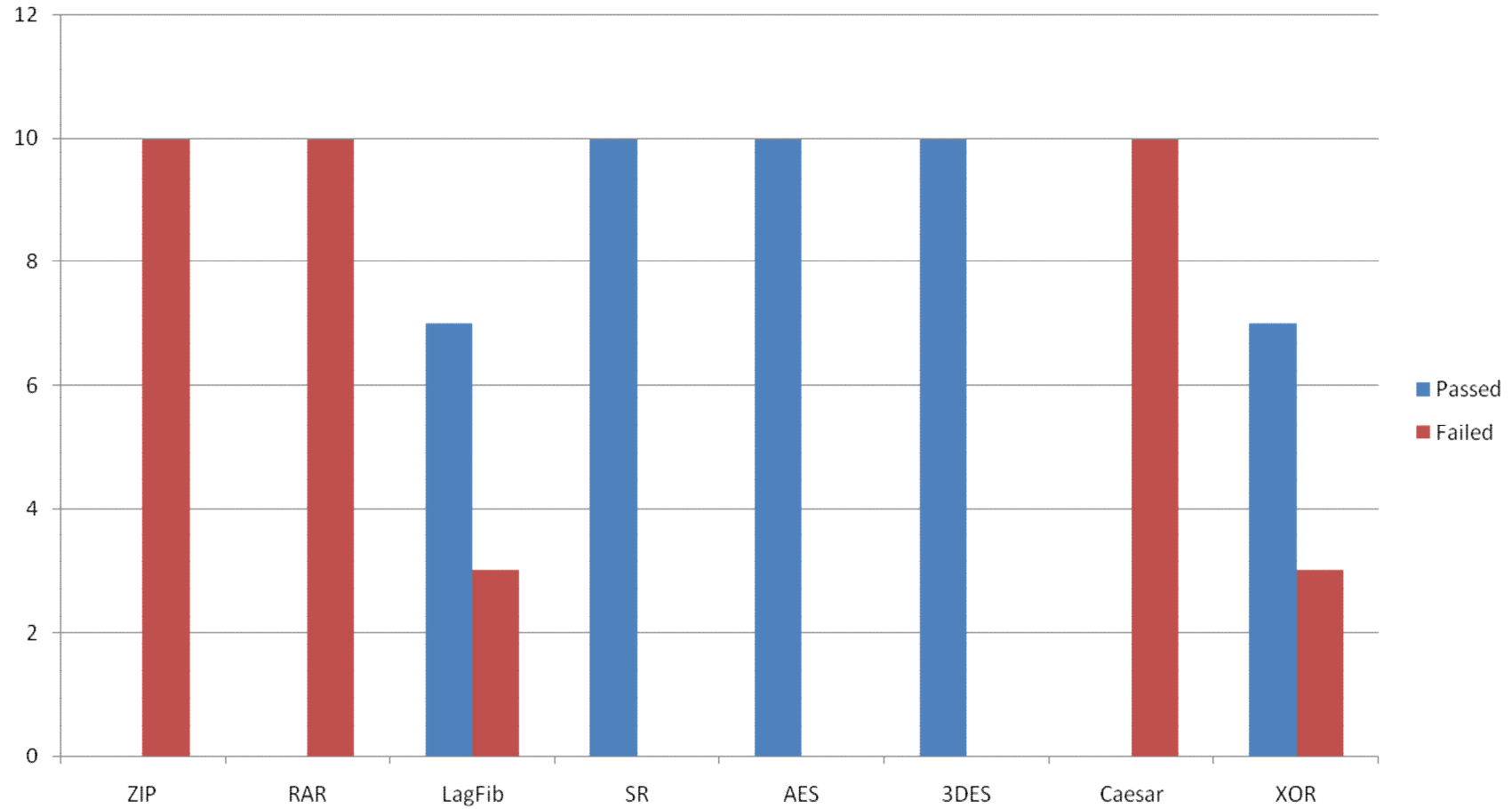
Overlapping Template



Binary Matrix Rank



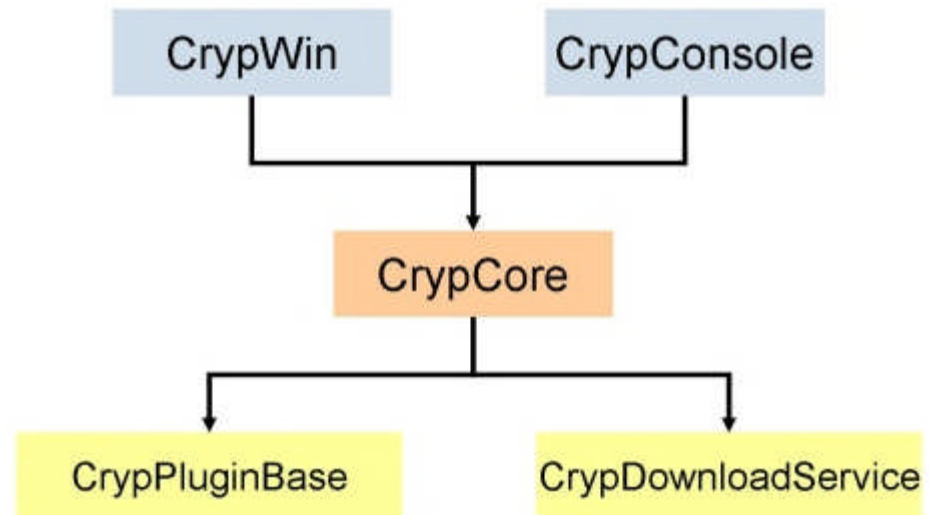
Maurer's Universal Test



Performance

Size	Frequency	Block Freq	Runs	Longest Run of ones	Binary Matrix	Overlapping	Non Overlapping	Universal
1MB	15ms	30ms	78ms	46ms	265ms	203ms	4281ms	218ms
10MB	78ms	78ms	312ms	187ms	2218ms	1500ms	42sec	1640ms
100MB	656ms	703ms		1937ms	22 sec	14sec	6 min 4 sec	16 sec

Cryptool



Next Steps

- Finish the rest of the application
- Integrate into Cryptool2

Thank You

Questions ?