



GLI®

World Headquarters

600 Airport Road
Lakewood, NJ 08701
Phone (732) 942-3999
Fax (732) 942-0043
www.gaminglabs.com

Worldwide Locations

World Headquarters
Lakewood, New Jersey

U.S. Regional Offices
Colorado
Nevada

International Offices
GLI Africa
GLI Asia
GLI Australia Pty Ltd
GLI Austria GmbH
GLI Europe BV
GLI Italy
GLI South America

14 February 2025

Prizes and Betting Secretariat of the Ministry of Finance
Esplanade of the Ministries,
Block P, Civic-Administrative Zone
Brazil-DF, 70048901

RE: Random Number Generator Report

File Number: RN-711-PLF-24-01, Update

Dear Sir/Madam,

Enclosed, please find a detailed explanation of the Random Number Generator (RNG) testing results of the Prolific Trade N.V. 'Blaze RNG', for use with the "Crash", "Double", "Slide", "Fortune Double", "Mines", "Fruit Slice", and "Tower" games., evaluated against the applicable RNG-specific requirements listed herein.

Please visit Gaminglabs.com to view the applicable Terms and Conditions and GLI Product Certification Scheme.

Registration Number of Accreditation applicable to this Report:	A2LA 2428.05
---	--------------

If you should have any questions regarding this Random Number Generator Report, please feel free to contact our office.

Sincerely,

GAMING LABORATORIES INTERNATIONAL, LLC

Christine M. Gallo
Senior Vice President, Quality and Technical Compliance

mra

RANDOMNESS REPORT FOR THE BLAZE RNG

The intent of this report is to indicate that **Gaming Laboratories International, LLC (GLI)** has completed its evaluation of the Blaze random number generator (RNG), provided by Prolific Trade N.V.

SECTION I - SCOPE OF TESTING

GLI was provided the required materials to conduct a randomness evaluation on the Blaze RNG. The scope of this evaluation was limited to software verification, source code review, and statistical testing. The RNG was tested for its ability to randomly produce outcomes for the “Crash”, “Double”, “Slide”, “Fortune Double”, “Mines”, “Fruit Slice”, and “Tower” games.

The Blaze RNG was evaluated against the RNG-specific requirements of the following technical standards:

- LAW Brazil (Federal) Law No. 13,765, of December 17, 2018
- O722 Brazil (Federal) SPA/MF Ordinance No. 722, of May 2, 2024
- O1207 Brazil (Federal) SPA/MF Ordinance No. 1,207, of July 29, 2024

SECTION II - SOFTWARE VERIFICATION

Verify+ by Kobetron™ signatures for the Blaze RNG are as follows: Note that the first entry in this table contains the RNG itself, while the remaining entries contain the calculation logic code for each individual game. Partial relative paths are included, as some files have identical names that need to be differentiated.

File	Type	Signature
provably-fair/roll-service.js	Kobe4	C920
	MD5	3EE3ABB67BB4634B9CCE843A09F374A8
	SHA-1	60051E5A102785D6ADC17094440F251F2638DA47
Crash/calculation-logic.js	Kobe4	A579
	MD5	3699CF1B5503972A6DC3795338BC9CC8
	SHA-1	3C993664211358CE4056EBC10C31E10D65C9B353
Double/get-roll-from-hash.js	Kobe4	H8F1
	MD5	8CF2AEE5FEEC2C1C669AF7168C3206EF
	SHA-1	780DE094D4C74519DC7DE654541A383F769F8FAB

File	Type	Signature
slide (Blaze-Jonbet)/calculation-logic.js	Kobe4	A957
	MD5	4A05460D175BC9FDF84B9077C2B5DF14
	SHA-1	37F8FB03DD3CFD4E697E02E0467267BAEB283BF0
Fortune-double/get-roll-from-hash.js	Kobe4	46CU
	MD5	B277A4B7B5CE04C43A6A95AA91B657CC
	SHA-1	877C7E588E779C6016C0D7BFE50E01DAAB2401B5
Mines/mines-board.js	Kobe4	8327
	MD5	F0644A0CBCA36F670307A3217DE7F43A
	SHA-1	F8C9B785A686DF91C4A95EFF4B39D6B23C853EAE
Fruit-slice/fruitSlice-utils.js	Kobe4	8AH7
	MD5	89D9D76E0162656ACF18D63C2641A111
	SHA-1	9E8BA385C0D7FEF5946FDC5FE79F43EB87AA2515
Tower/tower-utils.js	Kobe4	86H9
	MD5	02879119FE3F6A71C6FC35AD017F09C7
	SHA-1	76C852F26706044A77A277D581748EFC EE368AF0

Table 1. Digital Signatures

SECTION III - SOURCE CODE REVIEW

GLI received the appropriate documentation and full source code which pertains to the generation of random numbers. GLI reviewed the source code provided by tracing the path of the RNG application from the initiation of the draw to the selected output of random numbers. GLI inspected the source code, where practicable, in an attempt to find any undisclosed switches or parameters having a possible influence on randomness and fair play. GLI assessed the ability of the RNG to produce all numbers within the desired range.

SECTION IV - STATISTICAL TESTING

The RNG parameters tested are listed in Table 2. GLI performed a data format check on each data set listed in order to confirm that these parameters were correctly represented in the data analyzed.

A set of numbers is said to be drawn *with replacement* if a number can be selected multiple times within the same draw. A set of numbers is said to be drawn *without replacement* if a number can only be selected once within the same draw.

Data Set	Range	Positions	Replacement
Crash - Instant Crash Trigger	0-14	1	N/A
Crash - Bonus Trigger	0-71	1	N/A
Double and Fortune Double	0-14	1	N/A
Mines and Fruit Slice	0-24	25	No
Tower – Two Columns	0-1	2	No
Tower – Three Columns	0-2	3	No
Tower – Four Columns	0-3	4	No

Table 2. RNG Parameters

In addition to final outcome data, GLI tested raw outcomes consisting of binary output from the main RNG algorithm prior to the application of any scaling algorithms. For a summary of the statistical tests applied to each data set, see *Appendix A*. For a description of the overall test methodology and a description of each test used, see *Appendix B*.

Overall, the RNG passed the battery of tests for each configuration at the 95%, 98%, and 99% confidence levels.

SECTION V - SUMMARY

Updates have been made to the previously issued report dated November 29, 2024, which include the addition of the games "Fortune Double", "Mines", "Fruit Slice" and "Tower" throughout the report, added the description of what a replacement draw is, updated Tables 1 and 2, and updated the Appendix A Table, as well as the description of the tests performed.

Overall Evaluation of the Random Number Generator

GLI's conclusion based upon the tests applied to the Blaze RNG data is that this RNG has exhibited random behavior and is suitable for the applications as described herein. If a game utilizes different RNG parameters than the ones listed in this report, the RNG should be resubmitted to test that set of parameters.

APPENDIX A: Statistical Test Summary

						Test Names													
						Runs	Serial Correlation	Interplay Correlation	Adjacency Max-Min	Adjacency High-Low	Adjacency Blocks	Coupon Collector	Duplicates	Overlaps	Permutation	Total Distribution	Total Distribution by Position	DieHard	NIST
Data Set	Range	Positions	Replacement	Draws															
Crash - Instant Crash Trigger	0-14	1	N/A	20,000,000	X	X						X	X	X		X			
Crash - Bonus Trigger	0-71	1	N/A	100,000,000	X	X						X	X	X		X			
Double and Fortune Double	0-14	1	N/A	20,000,000	X	X						X	X	X		X			
Mines and Fruit Slice	0-24	25	No	20,000,000	X	X	X	X	X	X	X	X	X	X	X	X	X		
Tower - Two Columns	0-1	2	No	1,000,000	X	X	X					X	X	X		X	X		
Tower - Three Columns	0-2	3	No	2,000,000	X	X	X		X			X	X	X	X	X	X		
Tower - Four Columns	0-3	4	No	5,000,000	X	X	X	X	X	X	X	X	X	X	X	X	X		
Binary			Not Applicable																X

Table A 1. Tests Applied

APPENDIX B: Test Descriptions

B.1 Definitions. The following terms apply to the below test descriptions. Randomness Device or Random Number Generator (RNG) output may be collected multiple numbers at a time. Each set of numbers is called a draw. Each individual number has a particular order within the *draw*. This is referred to as the number *position*.

B.2 Distribution Comparisons. Many of the tests compare an observed numerical distribution with an expected distribution. Unless otherwise specified, this is done by means of a statistical chi-square goodness-of-fit test. The value chi-square is computed in the standard way. If k is a possible value, o_k is the observed count of that value, and e_k is the expected count:

$$\chi^2 = \sum_k \frac{(o_k - e_k)^2}{e_k}$$

In the case where expected counts are too small for accurate use of the above formula, values are ‘binned’ together to ensure an appropriate minimum expected count. The resultant value for chi-square is compared against the distribution for the appropriate number of degrees of freedom. Unusually high (distribution mismatch) or unusually low (insufficient randomness) chi-square values can be causes for data failure.

B.3 Meta-testing. Evaluation of groups of p -values may include a meta-test for extremity of high or low p -values, a meta-test for frequency of high or low p -values, and a meta-test for uniformity of p -values, as appropriate.

B.4 Confidence Level. The statistical tests conducted by GLI are done at a particular *confidence level*. Common confidence levels used include 95%, 98%, and 99%, depending on jurisdictional requirements, and intended use of the RNG. High confidence level testing has low risk of mistakenly failing a good RNG, but higher risk of passing a bad RNG. Lower confidence level testing has increased power of detecting bad RNGs, while also increasing the risk of false failures of good RNGs. Specifically, the confidence level represents the probability that an ideal source of randomness would pass the testing. If an RNG passes statistical tests at a given confidence level, passage at all *higher* confidence levels is implied.

B.5 Tests. Some tests are only applicable to certain types of data. Some tests may be applied only to a portion of the data. Some tests may require that the data be parsed, binned, or otherwise transformed, as necessitated by data format.

Adjacency Blocks:

For each draw, the data is first sorted. Then the amount of contiguous blocks of numbers is counted. These statistics are then compared against the expected. For example, if a draw consists of the numbers

1, 5, 4, 2, 6, 9

the data would be sorted and separated into blocks. The resulting statistic would be 3.

Adjacency High-Low:

For each draw, the number of local extrema ('highs' and 'lows') in the data is recorded and compared with the expected distribution. These are also referred to as 'turning points'. For example, if a draw consists of the numbers

1, 3, 5, 7, 2, 9

there would be one local maximum (7) and one local minimum (2). The resulting statistic would be 2.

Adjacency Max-Min:

For each draw, the difference between the maximum and minimum values is calculated and recorded. This is compared with the expected theoretical distribution. For example, if a draw consists of the numbers

2, 3, 6, 7, 4

the resulting statistic would be 5, the difference between the maximum value (7) and the minimum value (2).

Coupon Collector's:

The Coupon Collector's Test is applied positionally. The data is parsed until all possible values have been observed, then the number of values checked is recorded and the count is restarted. This is compared with the expected distribution. For example, if the set of all possible values is $\{0, 1, 2\}$ and the first position of each draw is

1, 0, 1, 0, 2, 0, 1, 2, ...

then all values are observed in the first position by the fifth draw. All values are then observed within the next 3 draws, so the first two statistics for the first position would be 5 and 3.

DieHard:

The DieHard Battery of Tests is a standard assessment of the randomness in raw outcomes generated from an RNG. The collection, designed by George Marsaglia, tests for a variety of patterns in the individual binary bits of RNG output. GLI uses a custom implementation to conduct DieHard testing.

Duplicates:

The Duplicates Test counts the number of times a draw is exactly duplicated in the data. In the case that a particular draw is repeated more than twice, every possible way to generate a duplicate is counted. This is compared against the theoretical distribution to verify that the number of duplicate draws falls within expected bounds. For example, consider the dataset consisting of the following draws of two numbers each.

- a) 1, 3
- b) 4, 1
- c) 1, 3
- d) 1, 3
- e) 4, 1
- f) 3, 1

The duplicate pairs are (a, c) , (a, d) , (c, d) , and (b, e) , for a total of 4 duplicates. (f) is not counted as a duplicate since the draw must match in order as well as values.

Interplay Correlation:

The Interplay Correlation Test measures statistical correlation between different positions of the same draw. For each pair of positions, statistical correlation is calculated as in the Serial Correlation Test. In the case of without replacement data, an adjustment is made to account for the expected resulting negative correlation.

Overlaps:

The Overlaps Test compares consecutive draws for overlapping values. The number of overlapping values is recorded for each pair of draws. This observed distribution of overlaps is then compared against the expected distribution. For example, if the following draws are observed consecutively,

a) 1, 4, 5, 6

b) 4, 1, 7, 6

the number of overlaps would be 3, representing the values 1, 4, and 6.

Permutation:

The Permutation Test is a test applicable to data that represents a reordering of numbers. Each draw can be considered as a permutation of the original ordering. Every permutation can be decomposed into disjoint cycles, which represent the possible positions a number would occupy if the same permutation is applied repeatedly. For each draw, three statistics are collected based on the cycle decomposition:

- The number of cycles.
- The size of the smallest cycle.
- The size of the largest cycle.

Each of these statistics generates a distribution of observations which are compared with their respective expected distributions. For example, if the following draw were observed as a reordering of the numbers from 1 to 6,

1, 3, 5, 4, 2, 6

the cyclic decomposition would be (1)(2 3 5)(4)(6). 1, 4, and 6 remain in their original positions, so they form their own cycles. The values 2, 3, and 5 are shuffled, so they form a single cycle together. The total number of cycles is 4, the smallest cycle has size 1, and the largest cycle has size 3.

Runs:

The Wald-Wolfowitz Runs Test is applied to each position within the draw. A center is established, typically the data median, and the number of 'runs' above and below the center are tallied. Values exactly equal to the center are discarded. This is compared to the expected distribution, which depends on the number of values above and below the center. For example, if the numbers drawn at a particular position were

2, 3, 1, 5, 4, 7, 3, 2, 3, 2, 3, 2, 6, 7, 3, 5

and the established center were the data median of 3, the data would be parsed for runs above 3 and runs below 3.

2, 3, 1, 5, 4, 7, 3, 2, 3, 2, 3, 2, 6, 7, 3, 5

This would be counted as 4 runs.

Serial Correlation:

The Serial Correlation Test measures statistical correlation between consecutive draws of the same position. For each position, the sample Pearson correlation coefficient is calculated. If X represents the first number, and Y the number that follows, then the coefficient is

$$r = \frac{cov(X, Y)}{s_X s_Y}$$

where s denotes the sample standard deviation. The coefficients are used to generate a p -value for each position.

Total Distribution:

The Total Distribution Test is a simple tally of all observed values throughout the data. This is compared with the expected distribution. Typically, the expected distribution is a uniform distribution. In the case of unequal weighting of values, an appropriate discrete distribution is used.

Total Distribution by Position:

The Total Distribution by Position Test tallies the observed distribution of values for each position within the draw. Each of these distributions is then compared with the expected.

NIST Test Suite

The NIST Test Suite comprises the following “bitwise” tests:

- Frequency (Monobits) Test
- Test for Frequency Within a Block
- Runs Test
- Test for The Longest Run of Ones in a Block
- Random Binary Matrix Rank Test
- Discrete Fourier Transform (Spectral) Test
- Non-Overlapping (Aperiodic) Template Matching Test
- Overlapping (Periodic) Template Matching Test
- Maurer’s Universal Statistical Test
- Linear Complexity Test
- Serial Test
- Approximate Entropy Test
- Cumulative Sums (Cumsum) Test
- Random Excursions Test
- Random Excursions Variant Test

Frequency (Monobits) Test

The focus of the test is the proportion of zeroes and ones for the entire sequence. The purpose of this test is to determine whether that number of ones and zeroes in a sequence are approximately the same as would be expected for a truly random sequence. The test assesses

the closeness of the fraction of ones to $\frac{1}{2}$, that is, the number of ones and zeroes in a sequence should be about the same.

Test for Frequency Within a Block Test

The focus of the test is the proportion of zeroes and ones within m -bit blocks. The purpose of this test is to determine whether the frequency of ones in an m -bit block is approximately $\frac{m}{s}$.

Runs Test

The focus of the test is the longest run of ones within m-bit blocks. The purpose of this test is to 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. Note that an irregularity in the expected length of the longest run of ones implies that there is also an irregularity in the expected length of the longest run of zeroes. Long runs of zeroes were not evaluated separately due to a concern about statistical independence among the tests.

Test for the Longest Run of Ones in a Block

The focus of the test is the longest run of ones within m-bit blocks. The purpose of this test is to 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. Note that an irregularity in the expected length of the longest run of ones implies that there is also an irregularity in the expected length of the longest run of zeroes. Long runs of zeroes were not evaluated separately due to a concern about statistical independence among the tests.

Random Binary Matrix Rank Test

The focus of the test is the rank of disjoint sub-matrices of the entire sequence. The purpose of this test is to check for linear dependence among fixed length substrings of the original sequence.

Discrete Fourier Transform (Spectral) Test

The focus of this test is the peak heights in the discrete Fast Fourier Transform. The purpose of this test is to 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.

Non-overlapping (Aperiodic) Template Matching Test

The focus of this test is the number of occurrences of pre-defined target substrings. The purpose of this test is to reject sequences that exhibit too many occurrences of a given non-periodic (aperiodic) pattern. For this test and for the Overlapping Template Matching Test, an m-bit window is used to search for a specific m-bit pattern. If the pattern is not found, the window slides one-bit position. For this test, when the pattern is found, the window is reset to the bit after the found pattern, and the search resumes.

Overlapping (Periodic) Template Matching Test

The focus of this test is the number of pre-defined target substrings. The purpose of this test is to reject sequences that show deviations from the expected number of runs of ones of a given length. Note that when there is a deviation from the expected number of ones of a given length, there is also a deviation in the runs of zeroes. Runs of zeroes were not evaluated separately due to a concern about statistical independence among the tests. For this test and for the Non-overlapping Template Matching Test, an m-bit window is used to search for a specific m-bit pattern. If the pattern is not found, the window slides one-bit position. For this test, when the pattern is found, the window again slides one bit, and the search is resumed.

Maurer's Universal Statistical Test

The focus of this test is the number of bits between matching patterns. The purpose of the test is to detect whether or not the sequence can be significantly compressed without loss of information. An overly compressible sequence is considered to be non-random.

Linear Complexity Test

The focus of this test is the length of a generating feedback register. The purpose of this test is to determine whether or not the sequence is complex enough to be considered random. Random sequences are characterized by a longer feedback register. A short feedback register implies non-randomness.

Serial Test

The focus of this test is the frequency of each and every overlapping m-bit pattern across the entire sequence. The purpose of this test is to 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. The pattern can overlap.

Approximate Entropy Test

The focus of this test is the frequency of each and every overlapping m -bit pattern. The purpose of the test is to compare the frequency of overlapping blocks of two consecutive/adjacent lengths (m and $m + 1$) against the expected result for a random sequence.

Cumulative Sum (Cumsum) Test

The focus of this test is the maximal excursion (from zero) of the random walk defined by the cumulative sum of adjusted $(-1, +1)$ digits in the sequence. The purpose of the test is to 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 random walk should be near zero. For non-random sequences, the excursions of this random walk away from zero will be too large.

Random Excursions Test

The focus of this test is the number of cycles having exactly K visits in a cumulative sum random walk. The cumulative sum random walk is found if partial sums of the $(0,1)$ sequence are adjusted to $(-1, +1)$. A random excursion of a random walk consists of a sequence of n steps of unit length taken at random that begin at and return to the origin. The purpose of this test is to determine if the number of visits to a state within a random walk exceeds what one would expect for a random sequence.

Random Excursions Variant Test

The focus of this test is the number of times that a particular state occurs in a cumulative sum random walk.

The purpose of this test is to detect deviations from the expected number of occurrences of various states in the random walk.