# Question - 1:

**MA 323 - Monte Carlo Simulation Assignment - 1**

## Vishisht Priyadarshi

**180123053**

* **Sequence of xi for a = 6, b = 0, m = 11 :**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sequence  Seeds(x0) | 1st  value | 2nd  value | 3rd  value | 4th  value | 5th  value | 6th  value | 7th  value | 8th  value | 9th  value | 10th  value |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 6 | 3 | 7 | 9 | 10 | 5 | 8 | 4 | 2 | 1 |
| 2 | 1 | 6 | 3 | 7 | 9 | 10 | 5 | 8 | 4 | 2 |
| 3 | 7 | 9 | 10 | 5 | 8 | 4 | 2 | 1 | 6 | 3 |
| 4 | 2 | 1 | 6 | 3 | 7 | 9 | 10 | 5 | 8 | 4 |
| 5 | 8 | 4 | 2 | 1 | 6 | 3 | 7 | 9 | 10 | 5 |
| 6 | 3 | 7 | 9 | 10 | 5 | 8 | 4 | 2 | 1 | 6 |
| 7 | 9 | 10 | 5 | 8 | 4 | 2 | 1 | 6 | 3 | 7 |
| 8 | 4 | 2 | 1 | 6 | 3 | 7 | 9 | 10 | 5 | 8 |
| 9 | 10 | 5 | 8 | 4 | 2 | 1 | 6 | 3 | 7 | 9 |
| 10 | 5 | 8 | 4 | 2 | 1 | 6 | 3 | 7 | 9 | 10 |

* When seed (x0) = 0, only 1 distinct value appears (0) which goes on repeating. For seeds (x0) = 1 to 10, 10 distinct values from 1 to 10 appear for each seed before the sequence starts repeating intself ,i.e, period length = m – 1 = 10.
* **Sequence of xi for a = 3, b = 0, m = 11 :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sequence  Seeds(x0) | 1st value | 2nd  value | 3rd  value | 4th  value | 5th  value |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 3 | 9 | 5 | 4 | 1 |
| 2 | 6 | 7 | 10 | 8 | 2 |
| 3 | 9 | 5 | 4 | 1 | 3 |
| 4 | 1 | 3 | 9 | 5 | 4 |
| 5 | 4 | 1 | 3 | 9 | 5 |
| 6 | 7 | 10 | 8 | 2 | 6 |
| 7 | 10 | 8 | 2 | 6 | 7 |
| 8 | 2 | 6 | 7 | 10 | 8 |
| 9 | 5 | 4 | 1 | 3 | 9 |
| 10 | 8 | 2 | 6 | 7 | 10 |

* When seed (x0) = 0, only 1 distinct value appears (0) which goes on repeating. For seeds (x0) = 1 to 10, 5 distinct values from 1 to 10 appear for each seed before the sequence starts repeating intself ,i.e, period length = 5.

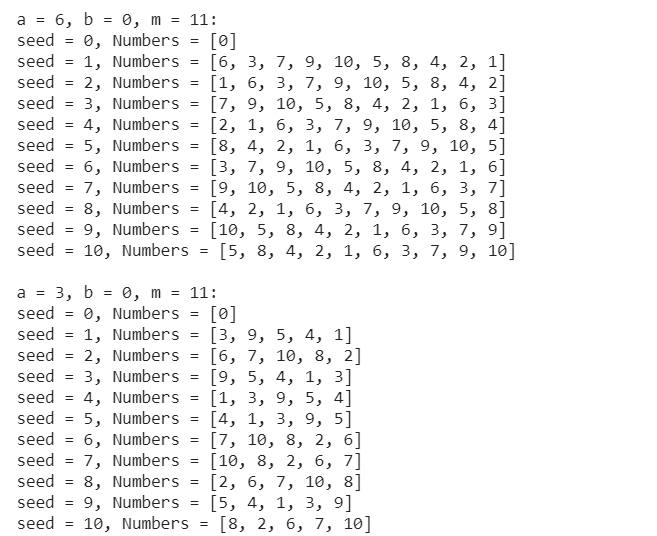
**Best Choice:**

* The largest possible period length of linear congruence generator is m – 1. This value is achieved when a = 6 (full period), while period length for a = 3 is only 5. So the linear congruence generator with a = 6 is preferred over a = 3 as it has higher period length.

This is because there will be more randomness in the generated numbers as there are more numbers in the sequence.

And x0 (seed) should be a non-zero value, as x0 = 0 has no randomness in it.

* **Output Screenshot:**

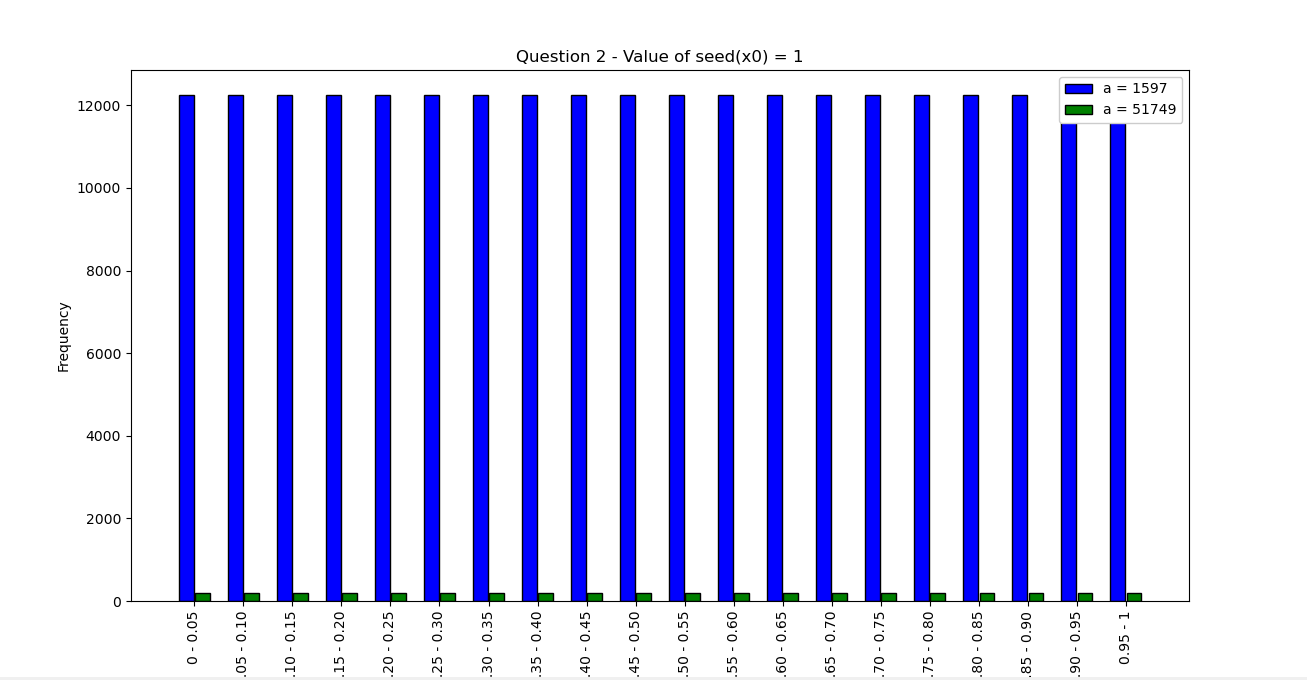
****

# Question - 2:

* **Tabulated Data of Frequency of numbers (ui) in different intervals:**

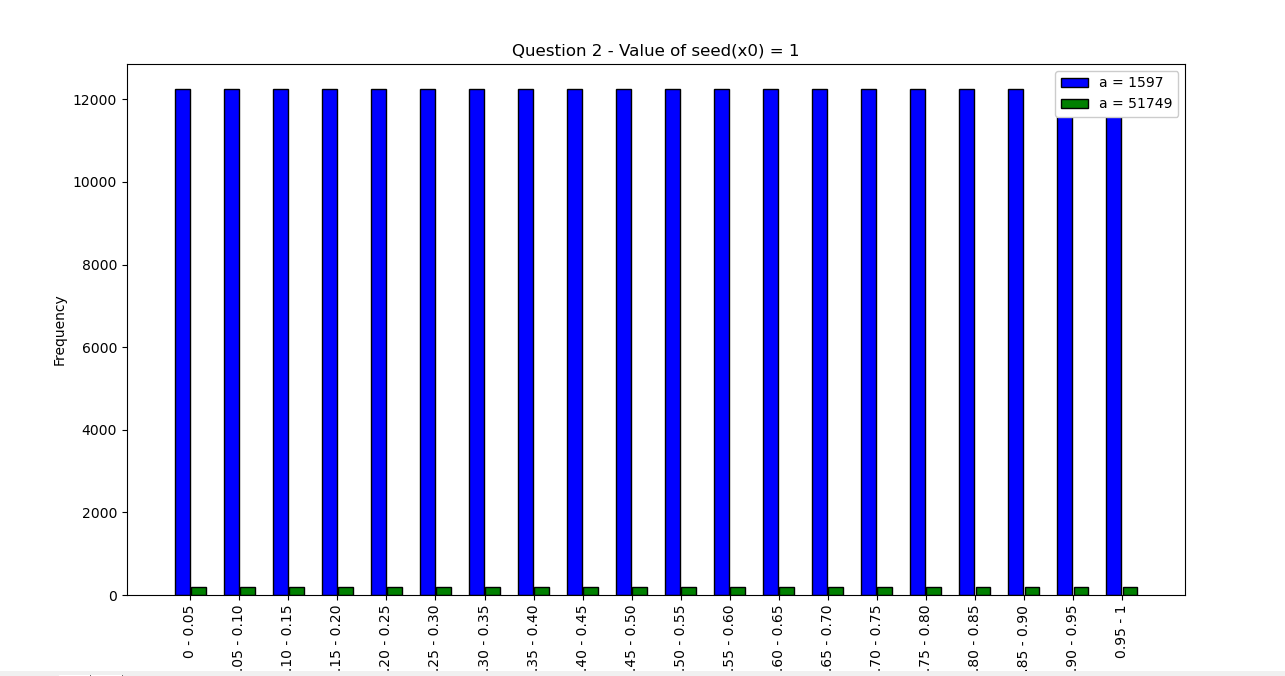
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | a = 1597, b = 1, m = 244944 | | | | | a = 51749, b = 1, m = 244944 | | | | |
| Seeds  Frequency | **X0 = 1** | **X0 = 2** | **X0 = 3** | **X0 = 4** | **X0 = 6** | **X0 = 1** | **X0 = 2** | **X0 = 3** | **X0 = 4** | **X0= 6** |
| 0.00 –  0.05 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 194 | 195 | 195 |
| 0.05 – 0.10 | 12247 | 12247 | 12247 | 12247 | 12247 | 194 | 194 | 193 | 194 | 194 |
| 0.10 – 0.15 | 12248 | 12248 | 12248 | 12248 | 12248 | 195 | 195 | 196 | 194 | 194 |
| 0.15 – 0.20 | 12247 | 12247 | 12247 | 12247 | 12247 | 193 | 193 | 193 | 195 | 194 |
| 0.20 – 0.25 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 196 | 194 | 195 |
| 0.25 – 0.30 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 194 | 195 | 195 |
| 0.30 – 0.35 | 12247 | 12247 | 12247 | 12247 | 12247 | 194 | 194 | 193 | 194 | 194 |
| 0.35 – 0.40 | 12247 | 12247 | 12247 | 12247 | 12247 | 194 | 194 | 196 | 194 | 194 |
| 0.40 – 0.45 | 12248 | 12248 | 12248 | 12248 | 12248 | 194 | 194 | 193 | 195 | 194 |
| 0.45 – 0.50 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 196 | 194 | 195 |
| 0.50 – 0.55 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 194 | 195 | 195 |
| 0.55 – 0.60 | 12247 | 12247 | 12247 | 12247 | 12247 | 194 | 194 | 193 | 194 | 194 |
| 0.60 – 0.65 | 12248 | 12248 | 12248 | 12248 | 12248 | 195 | 195 | 196 | 194 | 194 |
| 0.65 – 0.70 | 12247 | 12247 | 12247 | 12247 | 12247 | 193 | 193 | 193 | 195 | 194 |
| 0.70 – 0.75 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 196 | 194 | 195 |
| 0.75 – 0.80 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 194 | 195 | 195 |
| 0.80 – 0.85 | 12247 | 12247 | 12247 | 12247 | 12247 | 194 | 194 | 193 | 194 | 194 |
| 0.85 – 0.90 | 12247 | 12247 | 12247 | 12247 | 12247 | 194 | 194 | 196 | 194 | 194 |
| 0.90 – 0.95 | 12248 | 12248 | 12248 | 12248 | 12248 | 194 | 194 | 193 | 195 | 194 |
| 0.95 – 1.00 | 12247 | 12247 | 12247 | 12247 | 12247 | 195 | 195 | 196 | 194 | 195 |

* **Bar Diagram for a = 1597,51749, b = 1, m = 244944, seed (x0) = 1:**

****

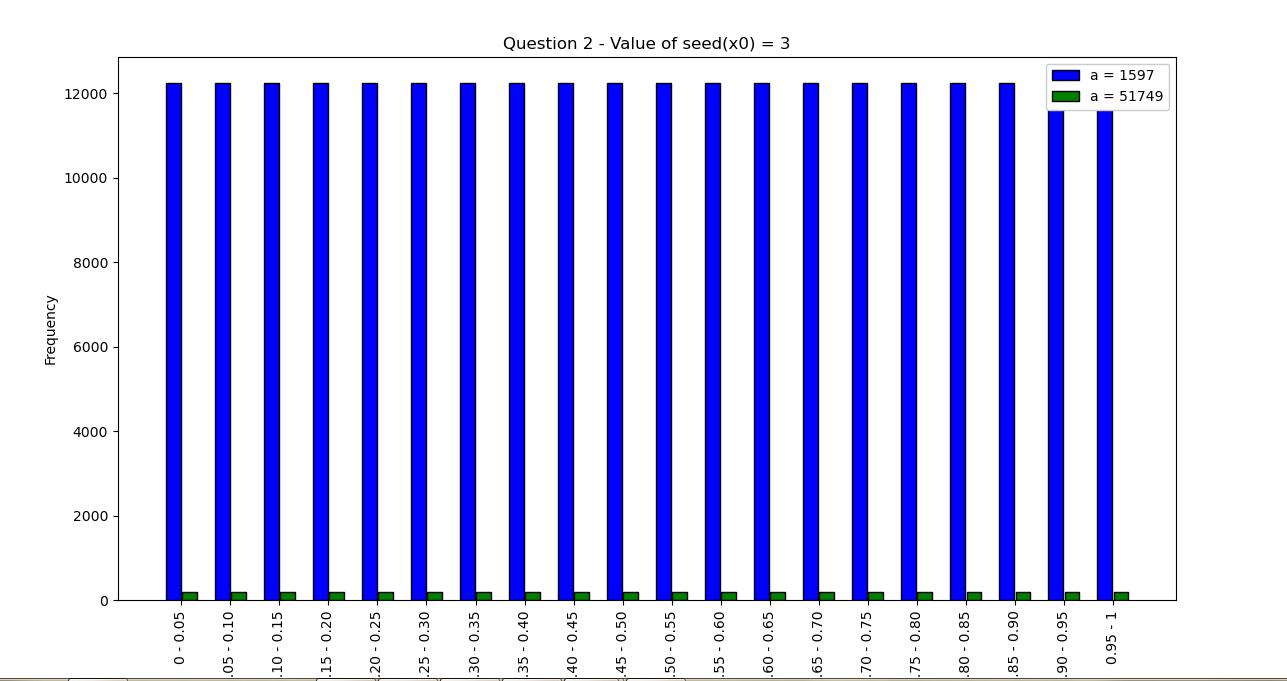
Frequency Range

* **Bar Diagram for a = 1597, 51749, b = 1, m = 244944, seed (x0) = 2:**



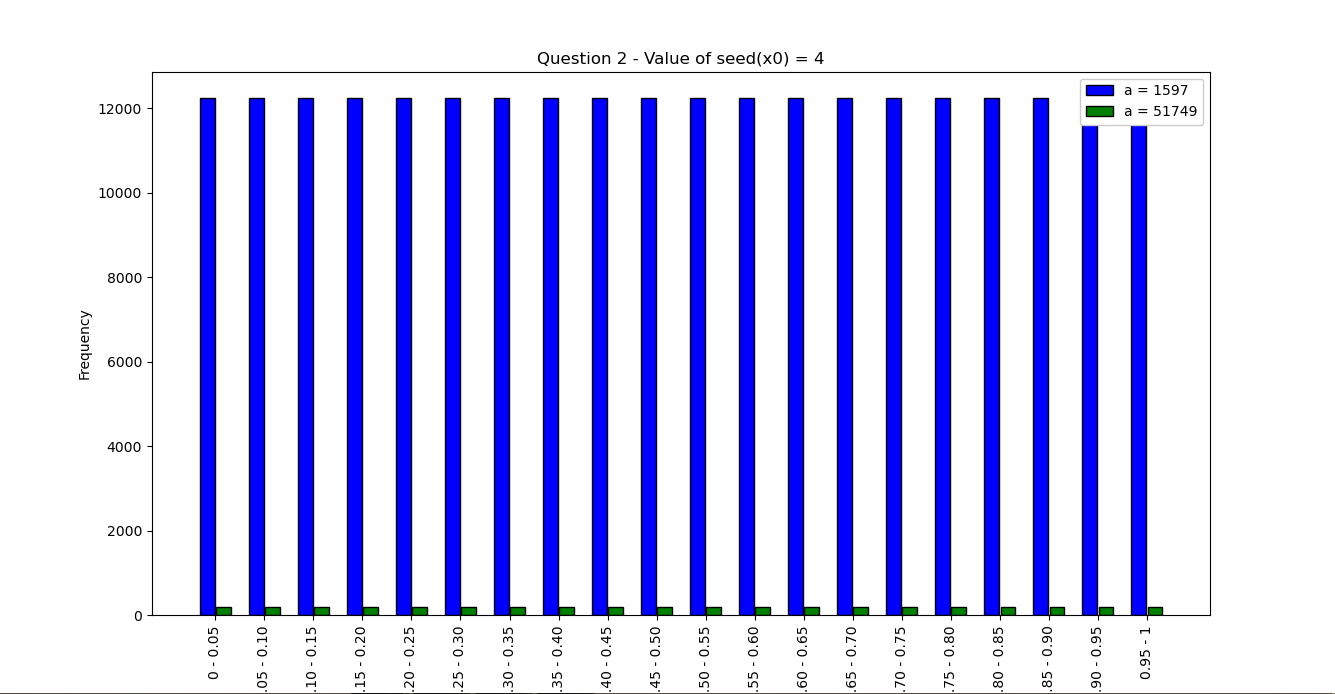
Frequency Range

* **Bar Diagram for a = 1597, 51749, b = 1, m = 244944, seed (x0) = 3:**

****

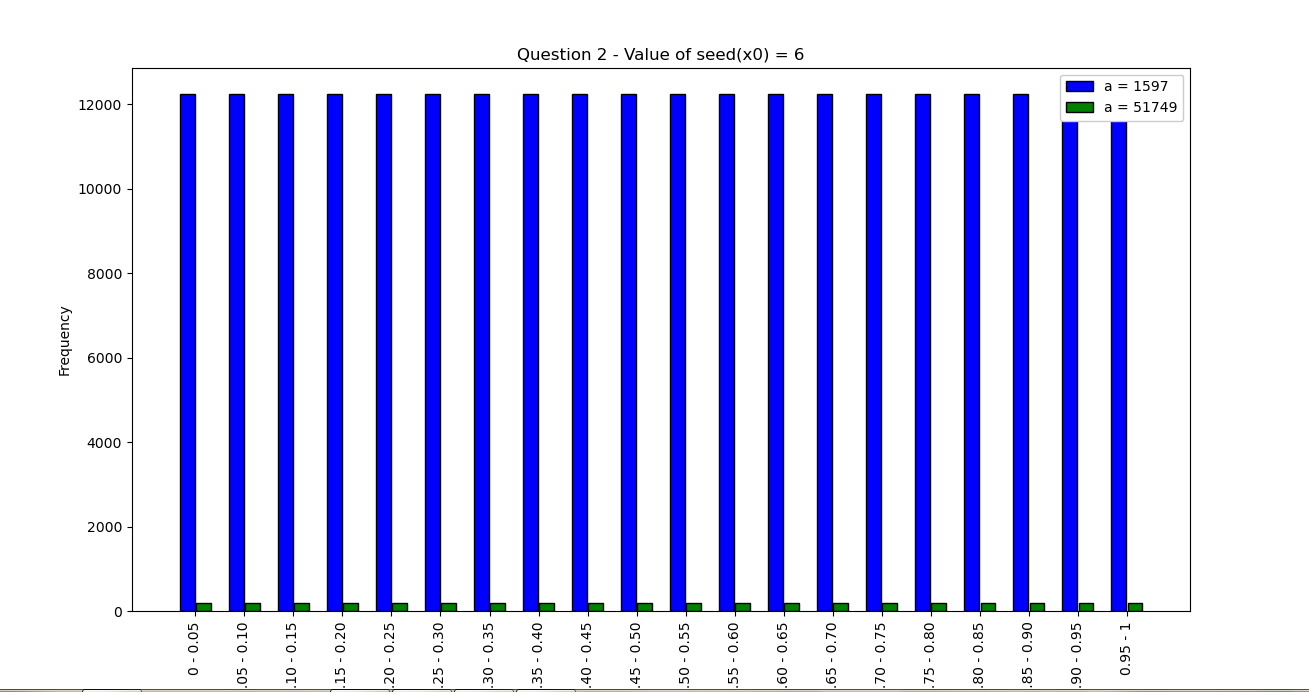
Frequency Range

* **Bar Diagram for a = 1597, 51749, b = 1, m = 244944, seed (x0) = 4:**

****

Frequency Range

* **Bar Diagram for a = 1597, 51749, b = 1, m = 244944, seed (x0) = 6:**

****

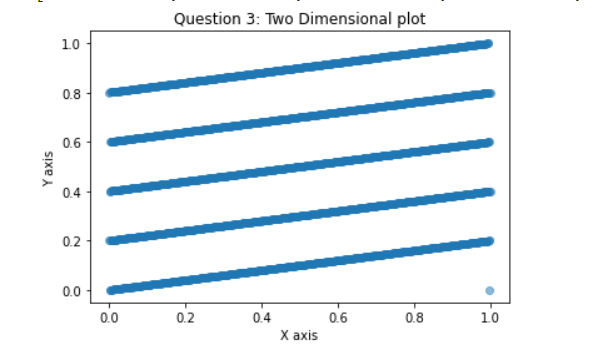
Frequency Range

* **Observations:**

1. The numbers are uniformly generated between 0 – 1. The frequency of different numbers lying in same length intervals are almost same. So, the random number generator follows the property of generation of numbers uniformly.
2. For different value of seed (x0), the frequencies are almost identical, and so the nature of bar graphs is same.
3. When a = 1597, b = 1, m = 244944, the Linear Congruence Generator has its full period, i.e. m – 1. But when a = 51749, b = 1, m = 244944, the Linear Congruence Generator does not achieve its full period.

# Question – 3 :

* **Scatter plot for co-ordinates (ui-1, ui) with a = 1229, b = 1, m = 2048 and seed(x0) = 1 :**



* **Observations:**

1. The scatter plot contains 5 almost parallel lines originating at different y – coordinates.
2. There is an outlier present at x = 1.0 (approx.). I believe this is present due to the precision issues while taking modulus in Python code (which is a bit different from the standard notion of modulus operation in other programming languages).
3. I found that this plot helps in what is known as “Spectral Test”. LCGs have a property that when plotted in 2 dimensions, lines will form, on which all possible outputs can be found.The spectral test compares the distance between these planes; the further apart they are, the worse the generator is.