# Tabular Data:

**MA 323 - Monte Carlo Simulation Assignment - 11**

## Vishisht Priyadarshi

**180123053**

The value of n was chosen to be 1000.

The simulation was run using two different LCGS:

1. Lagged Fibonacci generator (custom)
2. In-built LCG of python
3. **Lagged Fibonacci generator:**

Ui = (Ui – 17 – Ui – 5)

If Ui  < 0, set Ui  = Ui  + 1

The first 17 values were generated using following General Linear Congruence Generator:

xi + 1  = (a.xi + b) mod m

ui + 1 = xi + 1 / m

with a = 1229, b = 1, m = 2048 and seed (x0) = 1

|  |  |  |
| --- | --- | --- |
| **SI No.** | **N** | **Discrepancy** |
| **1** | 10 | 0.023000000000000007 |
| **2** | 20 | 0.012000000000000004 |
| **3** | 50 | 0.009999999999999998 |
| **4** | 100 | 0.009 |

**2. In-built LCG of python:**

|  |  |  |
| --- | --- | --- |
| **SI No.** | **N** | **Discrepancy** |
| **1** | 10 | 0.01899999999999999 |
| **2** | 20 | 0.013999999999999999 |
| **3** | 50 | 0.009000000000000001 |
| **4** | 100 | 0.009 |

The discrepancy generated using this method might fluctuate a little since the LCG was not seeded for getting more randomness in the produced uniform random variate.