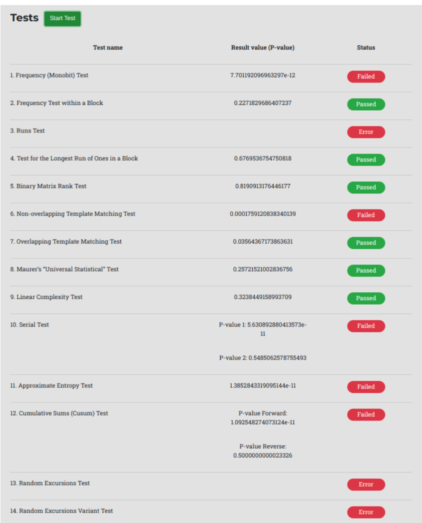
1 Testy NIST

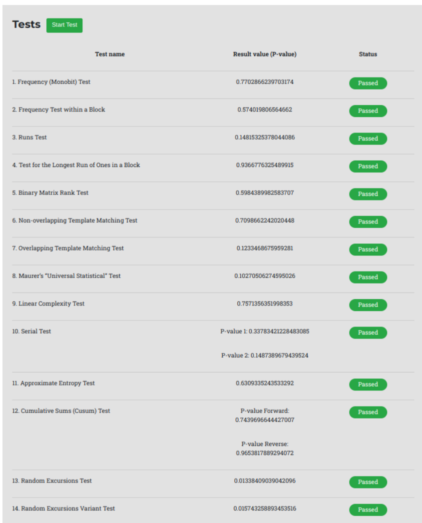
Porównano 3 metody generowania ciągów losowych bitów: Linear Congruential Generator, Mersenne Twister Generator oraz output metody haszującej SHA-1. W serwisie Random Bitstream Tester(`https://mzsoltmolnar.github.io/random-bitstream-tester`) zostały przeprowadzone ich testy. Rezultaty zostały przedstawione poniżej w tabeli oraz na kolejnych zrzutach ekranu.

Test	Typ generatora		
	LCG	Mersenne Twister	SHA-1
`Frequency(Monobit) test`	`FAILED`	`PASSED`	`PASSED`
`Frequency test within a block`	`PASSED`	`PASSED`	`PASSED`
`Runs test`	`FAILED`	`PASSED`	`PASSED`
`Test for the longest run of ones in block`	`PASSED`	`PASSED`	`PASSED`
`Binary matrix rank test`	`PASSED`	`PASSED`	`ERROR`
`Non-overlapping template matching test`	`FAILED`	`PASSED`	`ERROR`
`Overlapping template matching test`	`PASSED`	`PASSED`	`ERROR`
`Maurer's "Universal Statistical" test`	`PASSED`	`PASSED`	`ERROR`
`Linear complexity test`	`PASSED`	`PASSED`	`ERROR`
`Serial test`	`FAILED`	`PASSED`	`ERROR`
`Approximate entropy test`	`FAILED`	`PASSED`	`PASSED`
`Cumulative sums(Cusums) test`	`FAILED`	`PASSED`	`PASSED`
`Random excursions test`	`FAILED`	`PASSED`	`ERROR`
`Random excursions variant test`	`FAILED`	`PASSED`	`ERROR`


Tests Start Test

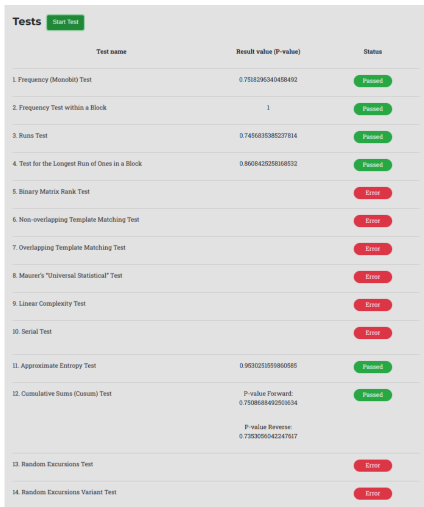
Test name	Result value (P-value)	Status
1. Frequency (Monobit) Test	7.701192096963297e-12	Failed
2. Frequency Test within a Block	0.2271829686407237	Passed
3. Runs Test		Error
4. Test for the Longest Run of Ones in a Block	0.6769536754750818	Passed
5. Binary Matrix Rank Test	0.8190913176446177	Passed
6. Non-overlapping Template Matching Test	0.0001759120838340139	Failed
7. Overlapping Template Matching Test	0.03564367173863631	Passed
8. Maurer's "Universal Statistical" Test	0.25721521002836756	Passed
9. Linear Complexity Test	0.3238449158993709	Passed
10. Serial Test	P-value 1: 5.630892880413573e-11 P-value 2: 0.5485062578755493	Failed
11. Approximate Entropy Test	1.3852843319095144e-11	Failed
12. Cumulative Sums (Cusum) Test	P-value Forward: 1.092548274073124e-11 P-value Reverse: 0.5000000000023326	Failed
13. Random Excursions Test		Error
14. Random Excursions Variant Test		Error

Figure 1: LCG


Tests Start Test

Test name	Result value (P-value)	Status
1. Frequency (Monobit) Test	0.7702866239703174	Passed
2. Frequency Test within a Block	0.574019806564662	Passed
3. Runs Test	0.14815325378044086	Passed
4. Test for the Longest Run of Ones in a Block	0.9366776325489915	Passed
5. Binary Matrix Rank Test	0.5984389982583707	Passed
6. Non-overlapping Template Matching Test	0.7098662242020448	Passed
7. Overlapping Template Matching Test	0.1233468675959281	Passed
8. Maurer's "Universal Statistical" Test	0.10270506274595026	Passed
9. Linear Complexity Test	0.7571356351998353	Passed
10. Serial Test	P-value 1: 0.33783421228483085 P-value 2: 0.1487389679439524	Passed
11. Approximate Entropy Test	0.6309335243533292	Passed
12. Cumulative Sums (Cusum) Test	P-value Forward: 0.7439696644427007 P-value Reverse: 0.9653817889294072	Passed
13. Random Excursions Test	0.01338409039042096	Passed
14. Random Excursions Variant Test	0.015743258893453516	Passed

Figure 2: Mersenne Twister


Tests Start Test

Test name	Result value (P-value)	Status
1. Frequency (Monobit) Test	0.7518296340458492	Passed
2. Frequency Test within a Block	1	Passed
3. Runs Test	0.7456835385237814	Passed
4. Test for the Longest Run of Ones in a Block	0.8608425258168532	Passed
5. Binary Matrix Rank Test		Error
6. Non-overlapping Template Matching Test		Error
7. Overlapping Template Matching Test		Error
8. Maurer's "Universal Statistical" Test		Error
9. Linear Complexity Test		Error
10. Serial Test		Error
11. Approximate Entropy Test	0.9530251559860585	Passed
12. Cumulative Sums (Cusum) Test	P-value Forward: 0.7508688492501634 P-value Reverse: 0.7353056042247617	Passed
13. Random Excursions Test		Error
14. Random Excursions Variant Test		Error

Figure 3: SHA1

Wnioski: Linear Congruential Generator nie ma własności dobrego generatora losowego. Używanie wyniku funkcji SHA-1 jako źródło losowych bitów także nie jest dobrym sposobem, ponieważ generuje ona jedynie 160 bitów. W obu pozostałych przypadkach do testów wprowadzano 1500000 bitów. Generator Mesenne Twister okazał się dobrym generatorem.

2 Błądzenie losowe na liczbach całkowitych - dystrybuanta

Kod użyty do przeprowadzenia symulacji w Matlabie:

```
SAMPLES_NUMBER = 100000;
N = 100;

VALS = zeros(1, SAMPLES_NUMBER);
for i = 1:SAMPLES_NUMBER
    VALS(i) = sum(2 * randi([0, 1], 1, N) - 1);
end

X_VALS = -N:N;
NORM_CDF = normcdf(X_VALS, 0, sqrt(N));

hold on;
cdfplot(VALS);
plot(X_VALS, NORM_CDF);

title("Dystrybuanta zmiennej losowej Sn dla N = " + N);
xlabel("t");
ylabel("P(Sn <= t)");
legend("Sn", "rozklad normalny");
```

Obserwacje:

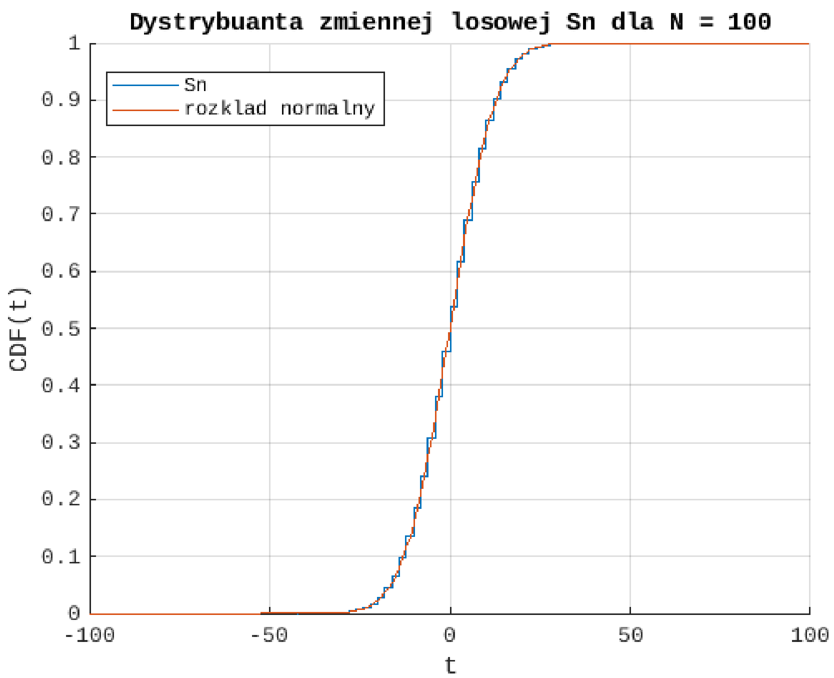


Figure 4: Porównanie rozkładu zmiennej S_n oraz rozkładu normalnego dla $N = 100$

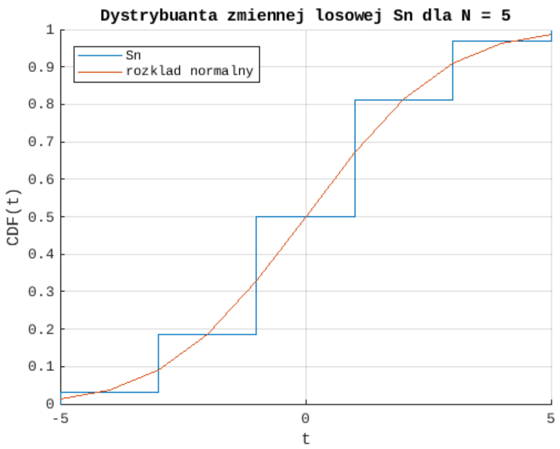


Figure 5: Porównanie rozkładu zmiennej S_n oraz rozkładu normalnego dla $N = 5$

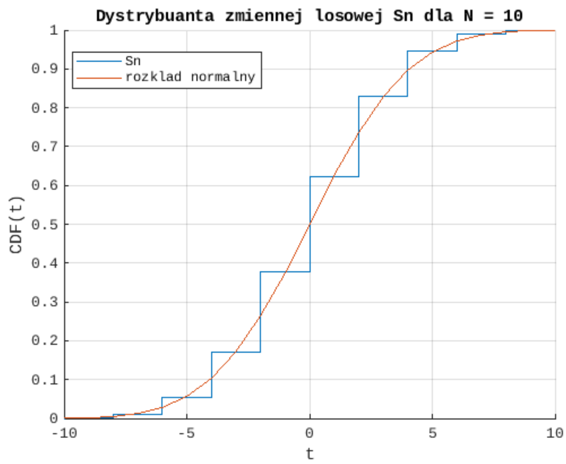


Figure 6: Porównanie rozkładu zmiennej S_n oraz rozkładu normalnego dla $N = 10$

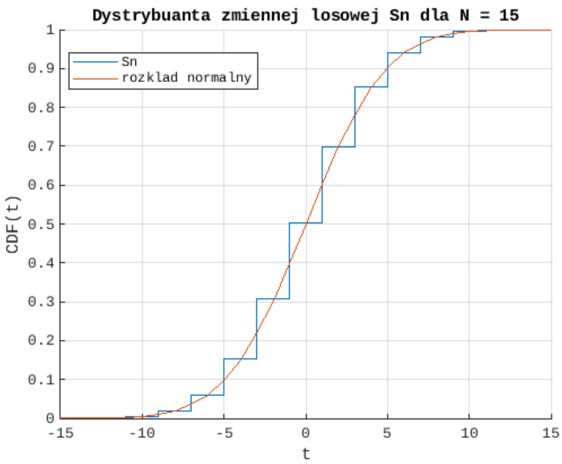


Figure 7: Porównanie rozkładu zmiennej S_n oraz rozkładu normalnego dla $N = 15$

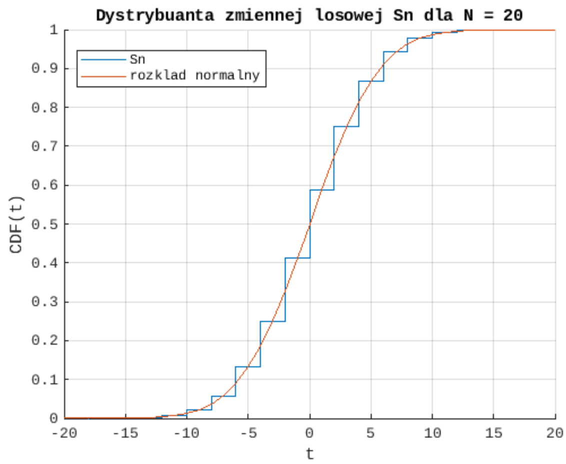


Figure 8: Porównanie rozkładu zmiennej S_n oraz rozkładu normalnego dla $N = 20$

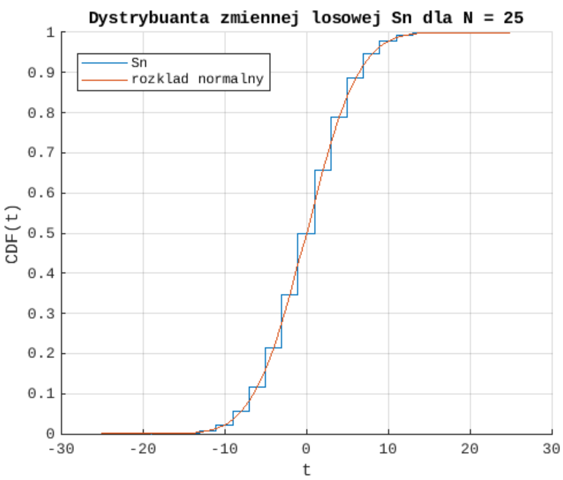


Figure 9: Porównanie rozkładu zmiennej S_n oraz rozkładu normalnego dla $N = 25$

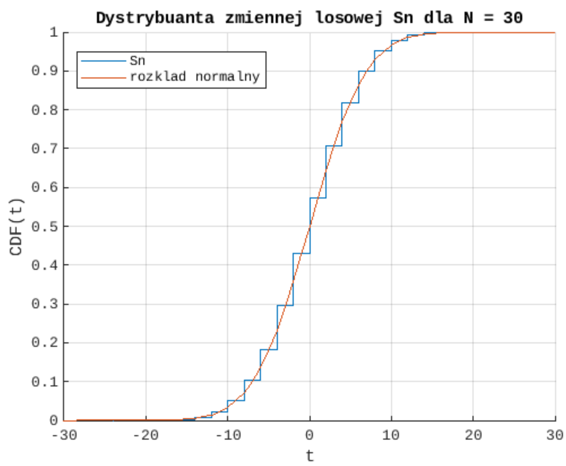


Figure 10: Porównanie rozkładu zmiennej S_n oraz rozkładu normalnego dla $N = 30$

3 Błądzenie losowe na liczbach całkowitych - rozkład czasu "nad osią OX"

Kod użyty do przeprowadzenia symulacji w Matlabie:

```
%Function calculating Pn
function Pn = random_walk(N)
    Xn = 2 * randi([0, 1], 1, N) - 1;
    Sn = [0, cumsum(Xn)];

    Dn = zeros(1, N);
    for i = 2:(N + 1)
        if Sn(i) > 0
            Dn(i) = 1;
        elseif Sn(i-1) > 0
            Dn(i) = 1;
        end
    end

    Ln = sum(Dn);
    Pn = Ln / N;
end

% Generating Chart
K = 5000;
N = 10000;

data = zeros(1, K);
for i = 1:K
    data(i) = random_walk(N);
end

hold on
histogram(data, 20, 'Normalization', 'pdf');

syms x
arc_dist = 1 / ( pi * sqrt(x * (1 - x)));
plt = fplot(arc_dist, [0, 1]);
plt.ShowPoles = false;

title("Gestosc rozkladu Pn");
xlabel("t");
ylabel("P(t)");
ylim([0, 5]);
legend("rozklad Pn", "rozklad arcsinus");
```

Obserwacje:

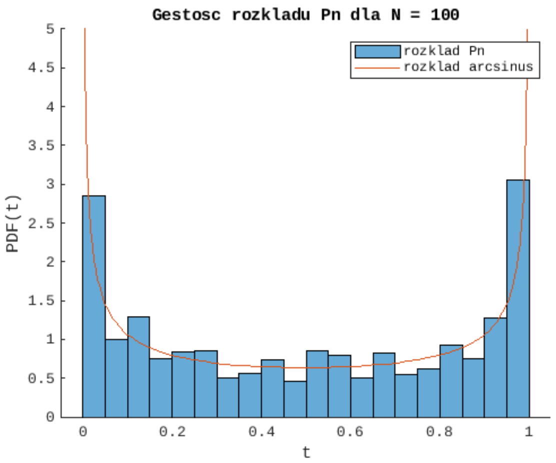


Figure 11: Porównanie gęstości rozkładu P_n oraz rozkładu arcsin dla $N = 100$

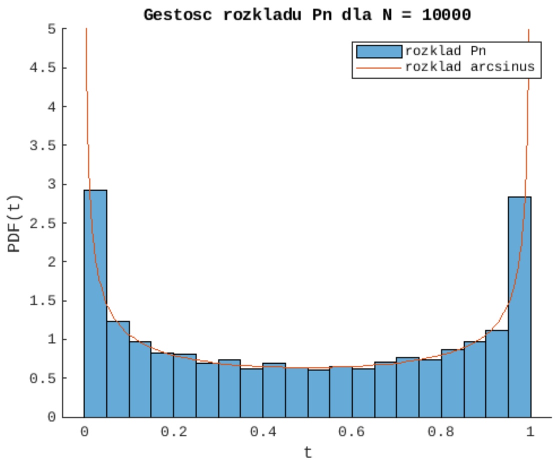


Figure 12: Porównanie gęstości rozkładu P_n oraz rozkładu arcsin dla $N = 1000$

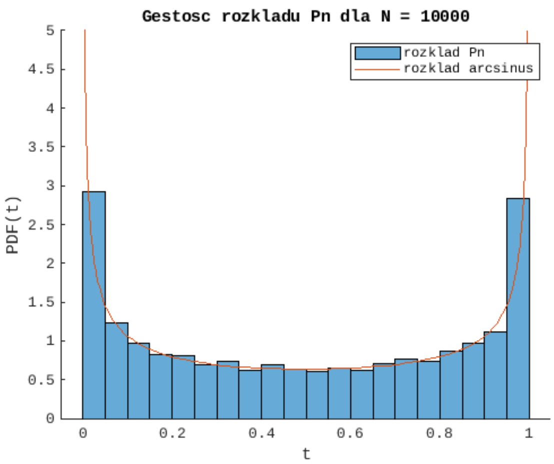


Figure 13: Porównanie gęstości rozkładu P_n oraz rozkładu arcsin dla $N = 10000$

4 Błądzenie losowe na liczbach całkowitych - wnioski

2.1. Wraz ze wzrostem wartości N wykres dystrybuantu zmiennej losowej S_n dąży do wykresu dystybuanty rozkładu normalnego ze średnią 0 oraz odchyleniem standardowym $\sqrt{n}$.

3.1. Wraz ze wzrostem wartości N wykres gęstości prawdopodobieństwa zmiennej P_n dąży do wykresu gęstości prawdopodobieństwa rozkładu arcus sinus.