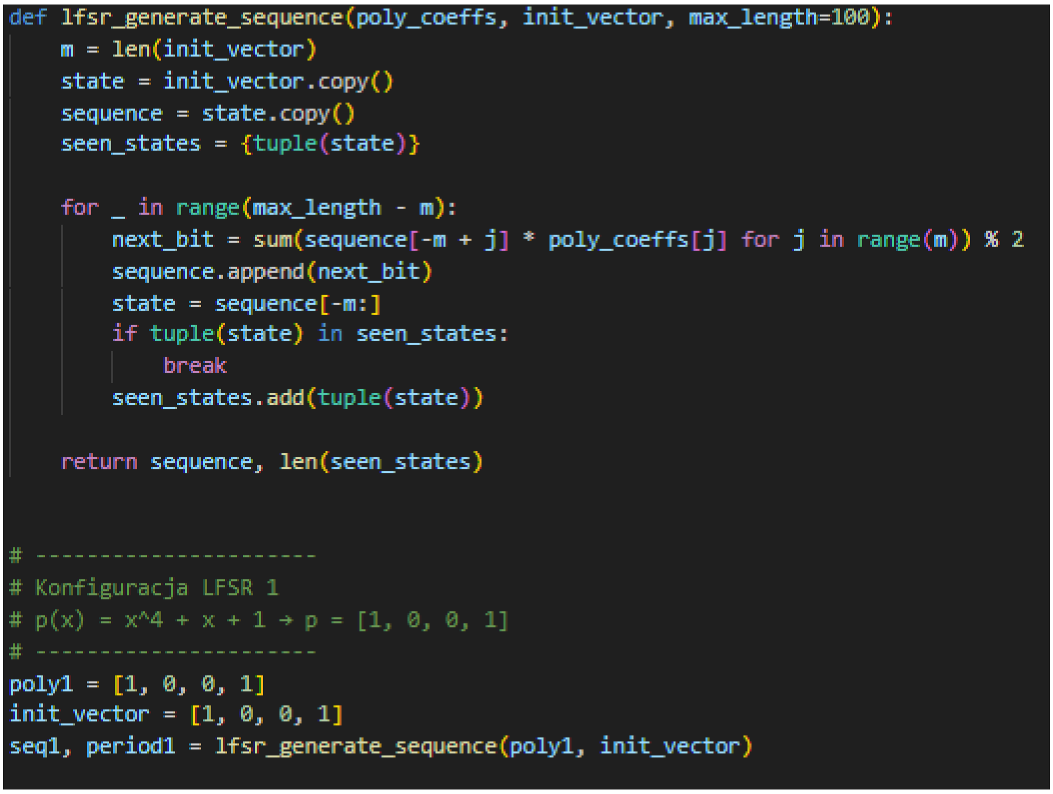
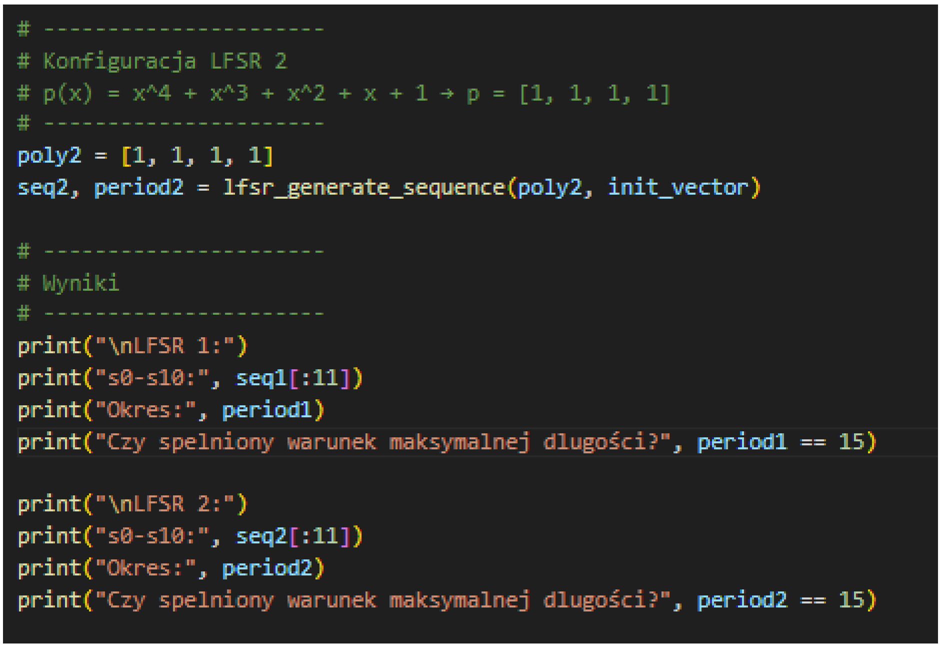
Lab 7 - kryptografia cd – LFSR

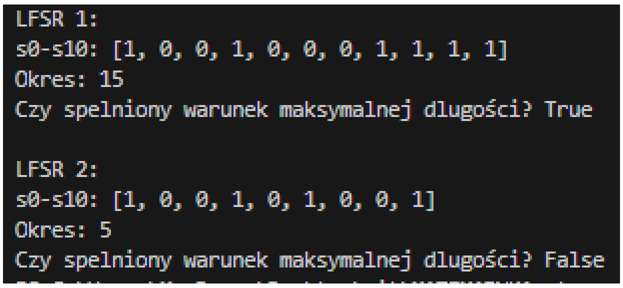
Arkadiusz Kurnik, Jan Cichoń



```python
def lfsr_generate_sequence(poly_coeffs, init_vector, max_length=100):
    m = len(init_vector)
    state = init_vector.copy()
    sequence = state.copy()
    seen_states = {tuple(state)}

    for _ in range(max_length - m):
        next_bit = sum(sequence[-m + j] * poly_coeffs[j] for j in range(m)) % 2
        sequence.append(next_bit)
        state = sequence[-m:]
        if tuple(state) in seen_states:
            break
        seen_states.add(tuple(state))

    return sequence, len(seen_states)


# ----------------------
# Konfiguracja LFSR 1
# p(x) = x^4 + x + 1 → p = [1, 0, 0, 1]
# ----------------------
poly1 = [1, 0, 0, 1]
init_vector = [1, 0, 0, 1]
seq1, period1 = lfsr_generate_sequence(poly1, init_vector)
```



```python
# ----------------------
# Konfiguracja LFSR 2
# p(x) = x^4 + x^3 + x^2 + x + 1 → p = [1, 1, 1, 1]
# ----------------------
poly2 = [1, 1, 1, 1]
seq2, period2 = lfsr_generate_sequence(poly2, init_vector)

# ----------------------
# Wyniki
# ----------------------
print("\nLFSR 1:")
print("s0-s10:", seq1[:11])
print("Okres:", period1)
print("Czy spelniony warunek maksymalnej dlugości?", period1 == 15)

print("\nLFSR 2:")
print("s0-s10:", seq2[:11])
print("Okres:", period2)
print("Czy spelniony warunek maksymalnej dlugości?", period2 == 15)
```



```
LFSR 1:
s0-s10: [1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1]
Okres: 15
Czy spelniony warunek maksymalnej dlugości? True

LFSR 2:
s0-s10: [1, 0, 0, 1, 0, 1, 0, 0, 1]
Okres: 5
Czy spelniony warunek maksymalnej dlugości? False
```