

```

# 1.Parameter
a = 5 # Konstanta pengali
m = 100 # Modulus (rentang nomor antrian 1 sampai 100)
Z0 = 42 # Seed (bilangan awal)
n = 20 # Jumlah nomor antrian yang akan dihasilkan

# 2.INISIALISASI DAFTAR
# Inisialisasi daftar untuk menyimpan hasil
results = []

# 3. ALGORITMA RNG
Z = Z0 # Bilangan awal
for i in range(1, n + 1):
    Z_prev = Z # Menyimpan bilangan sebelumnya
    Z = (a * Z) % m # Menghitung nomor antrian selanjutnya
    U = Z / m # Menghitung bilangan acak seragam
    rumus = f"({a} * {Z_prev}) mod {m}" # Rumus yang digunakan
    results.append([i, Z_prev, rumus, Z, round(U, 3)]) # Menyimpan hasil

# 4. OUTPUT HASIL
# Menampilkan tabel hasil pengacakan nomor antrian rumah sakit
print(f"{'No. Antrian':<12} | {'Nomor Sebelumnya':<18} | {'Rumus RNG':<30} | {'Nomor Antrian':<15} | {'U = Nomor/MOD (Acak Seragam)':<15}")
print("-" * 120)
for row in results:
    print(f"{row[0]:<12} | {row[1]:<18} | {row[2]:<30} | {row[3]:<15} | {row[4]:<15}")

```

No. Antrian	Nomor Sebelumnya	Rumus RNG	Nomor Antrian	U = Nomor/MOD (Acak Seragam)
1	42	(5 * 42) mod 100	10	0.1
2	10	(5 * 10) mod 100	50	0.5
3	50	(5 * 50) mod 100	50	0.5
4	50	(5 * 50) mod 100	50	0.5
5	50	(5 * 50) mod 100	50	0.5
6	50	(5 * 50) mod 100	50	0.5
7	50	(5 * 50) mod 100	50	0.5
8	50	(5 * 50) mod 100	50	0.5
9	50	(5 * 50) mod 100	50	0.5
10	50	(5 * 50) mod 100	50	0.5
11	50	(5 * 50) mod 100	50	0.5
12	50	(5 * 50) mod 100	50	0.5
13	50	(5 * 50) mod 100	50	0.5
14	50	(5 * 50) mod 100	50	0.5
15	50	(5 * 50) mod 100	50	0.5
16	50	(5 * 50) mod 100	50	0.5
17	50	(5 * 50) mod 100	50	0.5
18	50	(5 * 50) mod 100	50	0.5
19	50	(5 * 50) mod 100	50	0.5
20	50	(5 * 50) mod 100	50	0.5