```
(* c - jak w funkcji/zadaniu, K - S_x modulo K,
n - liczba testow, X - dane jak w zadaniu *)
c = 1 234 567;
K = 10^16;
n = 1000;
X = .;
x = 12 340;


target = OpenWrite["C:\\Users\\Tomek\\Desktop\\KolAlg001.txt"];
Write[target, c]
Write[target, K]
Write[target, n]
For[i = 1, i ≤ n, i++,
 Write[target, 10^18 * i^6/n^6]
]
Close[target];

Clear[F];
F[1] = 1;
F[2] = 1;
F[n_] := F[n] = c * F[n - 1] + F[n - 2];

Clear[S];
S[n_] := S[n] = Sum[F[i], {i, 1, n}];
x = 4096;
Mod[S[x], K]
```

5 837 822 004 875 987

```
Clear[ModT];
ModT[n_, k_] := If[n >= 0, Mod[n, k], Mod[n - 1, k] - k + 1, Mod[n, k]];
SetAttributes[ModT, Listable];
Clear[x1, x2, x3, start, step, i];
```

$$\text{start} = \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} /. \text{Solve}\left[\text{MatrixPower}\left[\begin{pmatrix} c & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, 2\right] . \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} == \begin{pmatrix} 2 \\ 1 \\ 2 - c \end{pmatrix}, \{x1, x2, x3\}\right][[1]];$$

$$\text{step}[0] = \begin{pmatrix} c & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix};$$

```
step[n_] := step[n] = ModT[MatrixPower[step[n - 1], 2], K];

For[i = 0, i < BitLength[x], i++,
  If[BitGet[x, i] == 1, start = ModT[step[i].start, K]]
 ];
Mod[start[[1]][[1]], K]
```

641 539 575 634 307

```
Timing[
 Clear[k, ModT];

 ModT[n_, k_] := If[n >= 0, Mod[n, k], Mod[n - 1, k] - k + 1, Mod[n, k]];
 SetAttributes[ModT, Listable];

 (* c - jak w funkcji/zadaniu, K - S_x modulo K,
 n - liczba testow, X - dane jak w zadaniu *)
 Clear[data, c, K, n, X, x];
 src = OpenRead["C:\\Users\\Tomek\\Desktop\\KolAlg001.txt"];
 data = ReadList[src, {Number}];
 Close["C:\\Users\\Tomek\\Desktop\\KolAlg001.txt"];
 target = OpenWrite["C:\\Users\\Tomek\\Desktop\\KolAlg001b.txt"];

 c = data[[1]][[1]];
 K = data[[2]][[1]];
 n = data[[3]][[1]];
 X = Take[data, {4, Length[data]}];


 For[j = 1, j ≤ n, j++,
   Clear[x1, x2, x3, start, step, i, x];
   x = X[[j]][[1]];
   start =
    {{x1}, {x2}, {x3}} /. Solve[MatrixPower[{{c, 1, 1}, {1, 0, 0}, {0, 0, 1}}, 2].{{x1}, {x2}, {x3}} == {{2}, {1}, {2 - c}}, {x1, x2, x3}][[1]];
   step[0] = {{c, 1, 1}, {1, 0, 0}, {0, 0, 1}};
   step[n_] := step[n] = ModT[MatrixPower[step[n - 1], 2], K];

   For[i = 0, i < BitLength[x], i++,
    If[BitGet[x, i] == 1, start = ModT[step[i].start, K]]
   ];
   Write[target, Mod[start[[1]][[1]], K]]
  ]
  Close[target];
]
```

```
{5.546875, Null}
```

```
Timing[
 Clear[ ModT];

 ModT[n_, k_] := If[n >= 0, Mod[n, k], Mod[n - 1, k] - k + 1, Mod[n, k]];
 SetAttributes[ModT, Listable];

 Clear[F];
 F[1] = 1;
 F[2] = 1;
 F[n_] := F[n] = c * F[n - 1] + F[n - 2];

 Clear[S];
 S[1] = F[1];
 S[n_] := S[n] = Sum[F[i], {i, 1, n}];

 Clear[coefList, cl, initSol];
 coefList = FindLinearRecurrence[Table[S[n], {n, 20}]];
 cl = Length[coefList];
 initSol = Reverse[Table[S[i], {i, 1, cl}]];

 Clear[changeMat, step];
 changeMat = Table[m_{i,j}, {i, cl}, {j, cl}];
 changeMat[[1]] = coefList;
 changeMat[[2 ;; cl, 1 ;; cl - 1]] = IdentityMatrix[cl - 1] ;
 changeMat[[2 ;; cl, cl]] = 0;
 step[0] = changeMat;
 step[n_] := step[n] = ModT[MatrixPower[step[n - 1], 2], K];

 Clear[V, v, start];
 V = Table[v_i, {i, 1, cl}];
 start = V /. Solve[MatrixPower[step[0], cl].V == initSol, V][[1]];

 For[i = 0, i < BitLength[x], i++,
  If[BitGet[x, i] == 1, start = ModT[step[i].start, K]]
 ];
 Mod[start[[1]], K]
]
```