

1. Each of the two pictures is a square with 100 points inside.

In both cases the coordinades of the points were calculated independently using a pseudorandom number generator. In one case an additional systematic modification of the coordinates was applied. The method of modification is unknown to us.

Which picture is the modified one? Formulate an argument which will support your guess.

2. It is easy to generate random numbers from the {1, 2, 3, 4, 5, 6} by throwing a dice. Suppose we have only one dice and we have to generate random integers in the interval [0, 10]. Describe the strategy of dice throwing which will generate each integer 0,1, ..., 10 with the same probability. (The dice is a classical 6-sided one).

3. There is an array of sorted integer values. Describe a strategy which will rearrange the values into a random order using a psaudorandom number generator. The method should work in a time proportional to the length of the array.

By rearranging into a random order we mean that all possible permutations of the values are equally likely.

4. Find out whether the lenght of the period of the given linear congruential generator is maximum possible.

A) *xn*+1 = (91 *xn* + 49) mod 600 C) *xn*+1 = (37 *xn* + 55) mod 144

B) *xn*+1 = (8 *xn* + 80) mod 49 D) *xn*+1 = (99 *xn* + 81) mod 113

5.Determine the period length in output of the Lehmer generator given by the relation

*xn*+1 = ((M−1)∙*xn*) mod M, (M is a prime).

6. Determine the upper and the lower bound of number of primes in the interval

A) [0, 109], B) [109, 2∙109], C) [2∙109, 3∙109].

7. We say that an integer as a quasi-prime if it is an integer power of a prime. Write a pseudo-code of a modification of Eratosthenes' sieve which will generate exactly all quasi-primes.

8. We say that an integer as a half-prime if it is a product of two primes. Write a pseudo-code of a modification of Eratosthenes' sieve which will generate exactly all half-primes.

9. A set {1000, 1001, …, 999999} was originally given. Then, all multiples of all primes less then 1000 (2, 3, 5, …, 991, 997) were excluded from S. Give an estimate of the cardinality of S and of the number of primes in S.

10. Determine the maximum number of primes in any of the intervals [30*k*, 30*k*+29], *k* = 1, 2, 3, 4, ... .

11. Find the numerical value of

A) GCD(220, 284), B) GCD , C) GCD(2100, 100!)

12. Example of modular exponentiation.

1889 mod 11 =

= 181011001B mod 11 =

= 181000000B + 10000B + 1000B + 1B  mod 11 =

= (181000000B ∙ 1810000B ∙181000B ∙181B ) mod 11 =

= ((181000000B mod 11) ∙ (1810000B mod 11) ∙ (181000B mod 11) ∙ (181B mod 11)) mod 11. (\*\*)

Intermediate calculations:

181B mod 11 = **7**

1810B mod 11 = ((181B mod 11) ∙ (181B mod 11)) mod 11 = (7∙7) mod 11 = 5

18100B mod 11 = ((1810B mod 11) ∙ (1810B mod 11)) mod 11 = (5∙5) mod 11 = 3

181000B mod 11 = ((18100B mod 11) ∙ (18100B mod 11)) mod 11 = (3∙3) mod 11 = 9

1810000B mod 11 = ((181000B mod 11) ∙ (181000B mod 11)) mod 11 = (9∙9) mod 11 = 4

18100000B mod 11 = ((1810000B mod 11) ∙ (1810000B mod 11)) mod 11 = (4∙4) mod 11 = 5

181000000B mod 11 = ((18100000B mod 11) ∙ (18100000B mod 11)) mod 11 = (5∙5) mod 11 = 3

Return to (\*\*):

((181000000B mod 11) ∙ (1810000B mod 11) ∙ (181000B mod 11) ∙ (181B mod 11)) mod 11 = (3∙4∙9∙7) mod 11 = 8.

Conclusion:

1889 mod 11 = 8.

13. Use the method presented in the previous example to compute the given values. In some cases, you may apply some reasoning to compute the result even faster:

A) 18189 mod 11 B) 2100 mod 20 C) 850 mod 7 D) 123456 mod 1000

14. The given code calculates integer power *xn*. Modify the code in such way that it will calculate *xn* mod *m*, for positive integer *m*.

BinPower(int *x*, int *n*) {

int *r* = 1, *y* = *x;*

while (*n* > 1) {

if (n % 2 == 1) *r* \*= *y;*

*y* \*= *y;*

*n* /= 2;

}

return *r\*y;*

}