Гончаренко Андрей Дмитриевич, 15 гр. 5 вариант

1) Тест Миллера-Рабина — вероятностный полиномиальный тест простоты. Тест Миллера-Рабина, наряду с тестом Ферма и тестом Соловея-Штрассена, позволяет эффективно определить, является ли данное число составным. Однако, с его помощью нельзя строго доказать простоту числа. Тем не менее тест Миллера-Рабина часто используется в криптографии для получения больших случайных простых чисел.
Как и тесты Ферма и Соловея-Штрассена, тест Миллера-Рабина опирается на проверку ряда равенств, которые выполняются для простых чисел. Если хотя бы одно такое равенство не выполняется, это доказывает что число составное.
Для теста Миллера-Рабина используется следующее утверждение:

> Пусть n — простое число и $n-1=2^s d$, где d — нечётно. Тогда для любого a из $\mathbb{Z}_n$ выполняется хотя бы одно из условий:
>
> 1. $a^d \equiv 1 \pmod{n}$
> 2. Существует целое число $r < s$ такое что $a^{2^r d} \equiv -1 \pmod{n}$

Результат выполнения программы:

```
/Users/andrey_pf/Desktop/workspace/KM/lab1/main/main/bin/Debug/net6.0/main
10
619
5

Process finished with exit code 0.
```

1. В первой строке мы вписываем размер простого числа в битах
2. Во второй строке получаем простое число
3. В третье строке получаем кол-во выполненных циклов

2) Тест Соловея-Штрассена — вероятностный тест простоты, открытый в 1970-х годах Робертом Мартином Соловеем совместно с Фолькером Штрассеном. Тест всегда корректно определяет, что простое число является простым, но для составных чисел с некоторой вероятностью он может дать неверный ответ. Основное преимущество теста заключается в том, что он, в отличие от теста Ферма, распознает числа Кармайкла как составные.
Алгоритм Соловея-Штрассена параметризуется количеством раундов k. В каждом раунде случайным образом выбирается число $a < n$. Если НОД$(a, n) > 1$, то выносится решение, что n составное. Иначе проверяется справедливость сравнения $a^{\left(\frac{n-1}{2}\right)} \equiv \left(\frac{a}{n}\right) mod\ n$. Если оно не

выполняется, то выносится решение, что n — составное. Если это сравнение выполняется, то a является свидетелем простоты числа n. Далее выбирается другое случайное a и процедура повторяется. После нахождения k свидетелей простоты в k раундах выносится заключение, что n является простым числом с вероятностью $1 - 2^{-k}$.

Вход: $n > 2$, тестируемое нечётное натуральное число;
k, параметр, определяющий точность теста.
Выход: *составное*, означает, что n точно составное;
вероятно простое, означает, что n вероятно является простым.

for $i = 1, 2, \ldots, k$:
a = случайное целое от 2 до $n-1$, включительно;
если НОД(a, n) > 1, **тогда**:
вывести, что n — составное, и **остановиться**.
если $a^{(n-1)/2} \not\equiv \left(\frac{a}{n}\right) \pmod{n}$, **тогда**:
вывести, что n — *составное*, и **остановиться**.

иначе вывести, что n — *простое с вероятностью* $1 - 2^{-k}$, и **остановиться**.

Результат выполнения программы:

```
/Users/andrey_pf/Desktop/workspace/KM/lab1/main/main/bin/Debug/net6.0/main
7
121
1

Process finished with exit code 0.
```

1. В первой строке мы вписываем размер простого числа в битах
2. Во второй строке получаем простое число
3. В третье строке получаем кол-во выполненных циклов

3) При заданном простом числе $p > 2$ тест позволяет за полиномиальное время от битовой длины p числа Мерсенна $M_p = 2^p - 1$ определить, является M_p простым или составным. Доказательство справедливости теста существенно опирается на функции Люка, что позволило обобщить тест Люка — Лемера на некоторые числа, вид которых отличен от чисел Мерсенна.

Пусть p — простое нечётное. Число Мерсенна $M_p = 2^p - 1$ простое тогда и только тогда, когда оно делит нацело $(p-1)$-й член последовательности

$$4,\ 14,\ 194,\ 37634,\ \ldots$$ [2],

задаваемой рекуррентно: $S_k = \begin{cases} 4 & k = 1, \\ S_{k-1}^2 - 2 & k > 1. \end{cases}$

Результат выполнения программы:

```
/Users/andrey_pf/Desktop/workspace/KM/lab1/main/main/bin/Debug/net6.0/main
Сгенерированная степень:
7
Сгенерированное число мерсена:
127

Process finished with exit code 0.
```