Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Основы теории чисел и их использование в криптографии

Студент: Вайсера Р.Л.

ФИТ 3 курс 4 группа

Преподаватель: Сазонова

Минск 2023

# Вычисление наибольшего общего делителя

Наибольшее целое число, которое делит без остатка числа *a* и *b*, называется наибольшим общим делителем этих чисел – НОД (*a, b*).

Наибольший общий делитель можно найти с помощью алгоритма Евклида, который заключается в том, чтобы от наибольшего числа из двух отнимать наименьшее, пока одно из них не станет равно нулю, а после вернуть наибольшее из двух число.

Для вычисления наибольшего общего делителя двух чисел реализована следующая функция, представленная на рисунке 1.1:

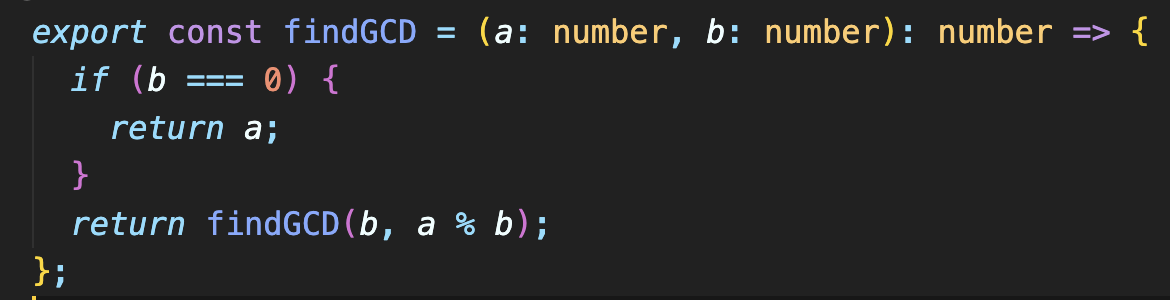


Рисунок 1.1 – Функция нахождения НОД двух чисел

# Поиск простых чисел

Для нахождения всех простых чисел не больше заданного числа *n* используется алгоритм, называемый «решетом Эратосфена». В соответствии с этим алгоритмом нужно выполнить следующие шаги:

1. выписать подряд все целые числа от двух (либо от *m*) до *n* (2, …, *n*). Пусть некоторая переменная (например, *s*) изначально равна 2, то есть первому простому числу;
2. удалить из списка числа от 2*s* до *n*, считая шагами по *s* (это будут числа, кратные *s*: 2*s*, 3*s*, 4*s*, …);
3. найти первое из оставшихся чисел в списке, большее чем *s*, и присвоить значению переменной *s* это число;
4. повторять шаги 2 и 3, пока возможно.

Для реализации данного алгоритма созданы следующие функции, представленные на рисунке 2.1:

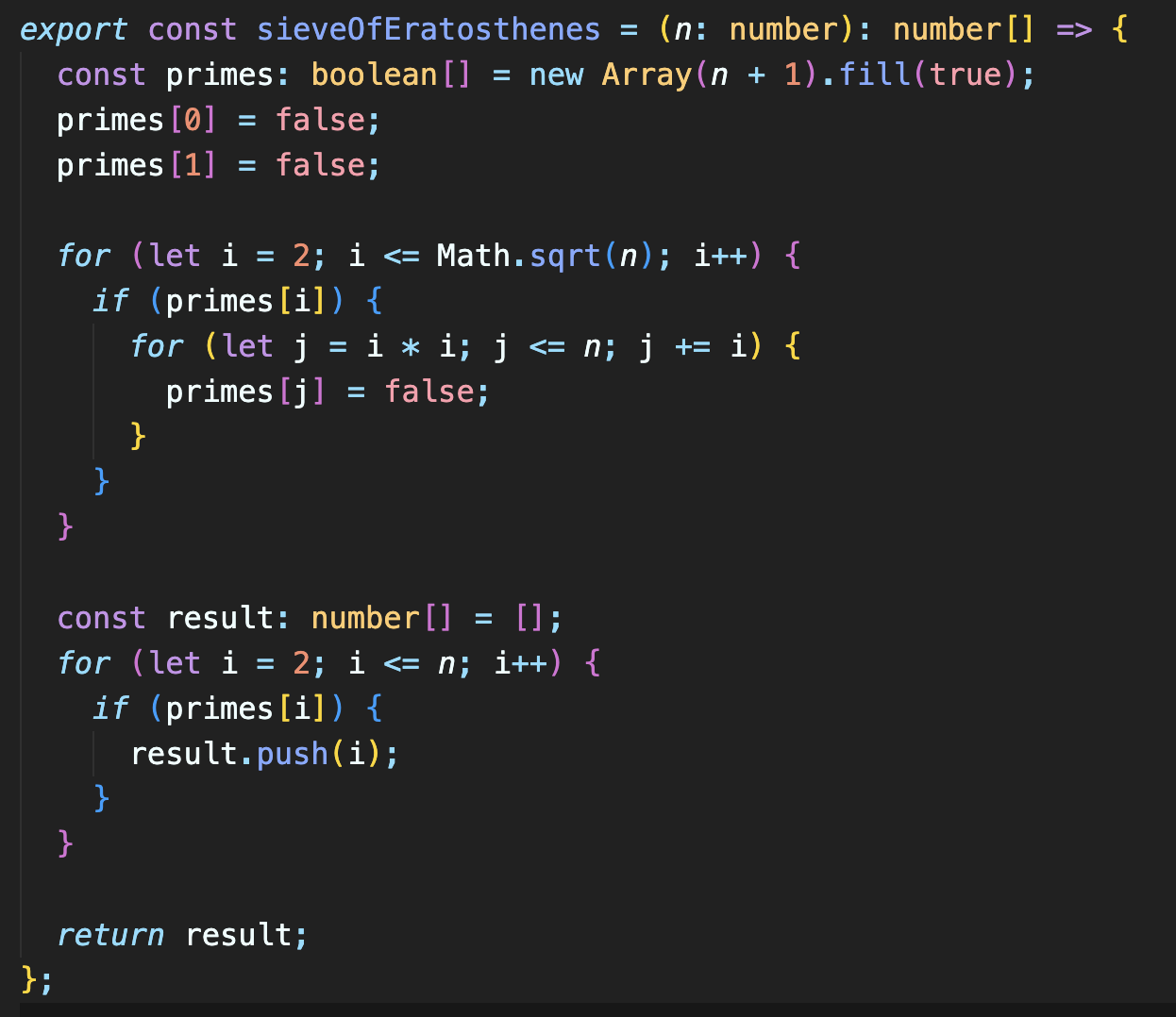


Рисунок 2.1 – Функция поиска простых чисел

Вывод данных функций для поиска простых чисел в интервалах [2, *n*] и [*m*, *n*], а также количество простых чисел в этих диапазонах и значение *n* / ln(*n*) представлены на рисунке 2.2.

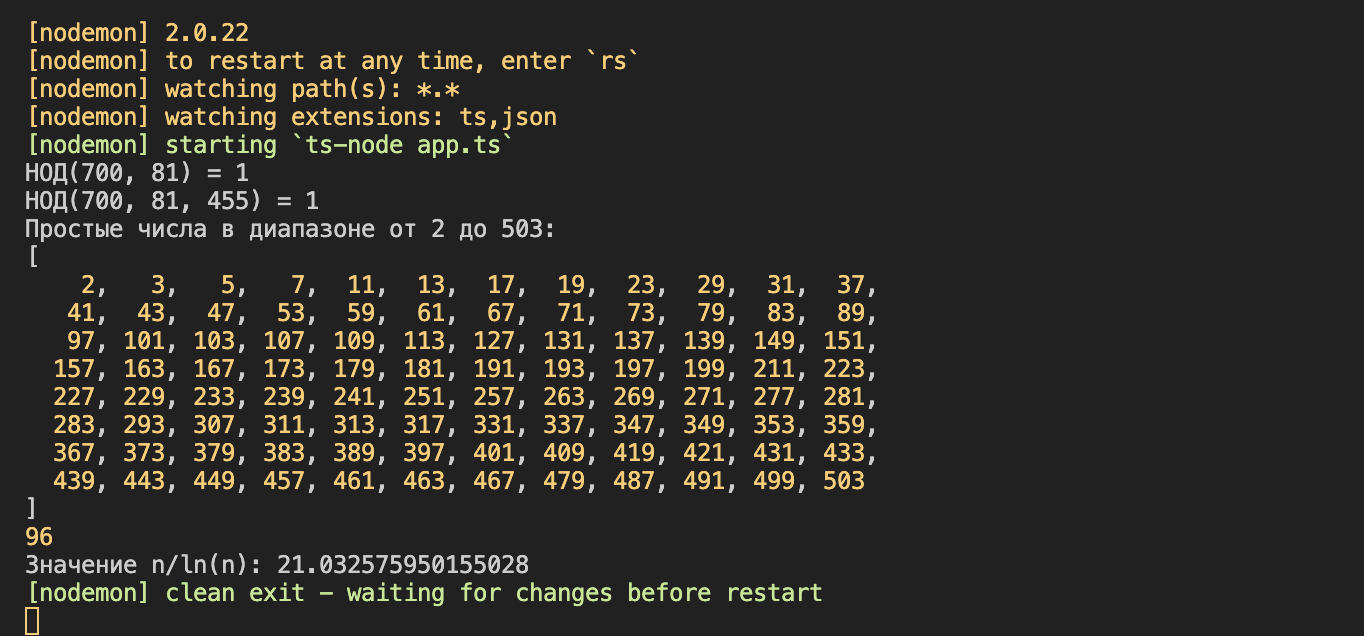


Рисунок 2.2 – Вывод функции поиска простых чисел

Выпишем числа от 450 до 503:

453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502.

Воспользуемся свойством 3 простых чисел и вычислим √503≈22.5, т. е. меньше 23.

Удалим последовательно все числа, делящиеся на простые числа от 2 до 20. Такими простыми числами являются: 2, 3, 5, 7, 11, 13, 17, 19.

Числа, которые делятся на 2: 454, 456, 458, 460, 462, 464, 466, 468, 470, 472, 474, 476, 478, 480, 482, 484, 486, 488, 490, 492, 494, 496, 498, 500, 502.

Оставшиеся числа: 453, 455, 457, 459, 461, 463, 465, 467, 469, 471, 473, 475, 477, 479, 481, 483, 485, 487, 489, 491, 493, 495, 497, 499.

Числа, которые делятся на 3: 453, 456, 459, 462, 465, 468, 471, 474, 477, 480, 483, 486, 489, 492, 495, 498.

Оставшиеся числа: 455, 457, 461, 463, 467, 469, 473, 475, 479, 481, 485, 487, 491, 493, 497, 499.

Числа, которые делятся на 5: 455, 460, 465, 470, 475, 480, 485, 490, 495, 500.

Оставшиеся числа: 453, 457, 461, 463, 467, 469, 471, 473, 477, 479, 481, 483, 487, 489, 491, 493, 497, 499.

Числа, которые делятся на 7: 455, 462, 469, 476, 483, 490, 497.

Оставшиеся числа: 453, 457, 461, 463, 467, 471, 473, 475, 479, 481, 485, 487, 491, 493, 495, 499.

Числа, которые делятся на 13: 457, 481, 493.

Оставшиеся числа: 453, 461, 463, 467, 471, 473, 475, 479, 485, 487, 491, 495, 499.

Числа, которые делятся на 17: 453, 471, 487, 493.

Оставшиеся числа: 461, 463, 467, 473, 475, 479, 485, 491, 495, 499..

Числа, которые делятся на 19: 475.

Оставшиеся числа: 457, 461, 463, 467, 479, 487, 491, 499, 503

# Каноническая форма записи числа

Любое число *n* можно представить в следующем виде, называемое канонической формой записи числа:

 (1.1)

где *p1*, *p2*,…, *pn* – разные простые множители числа, *a1*, *a2*, …, *an* – степени данных простых множителей. Также данные множители можно записывать подряд, по возрастанию, без использования степеней.

Для реализации представления числа в канонической форме реализована следующая функция, представленная на рисунке 3.1.

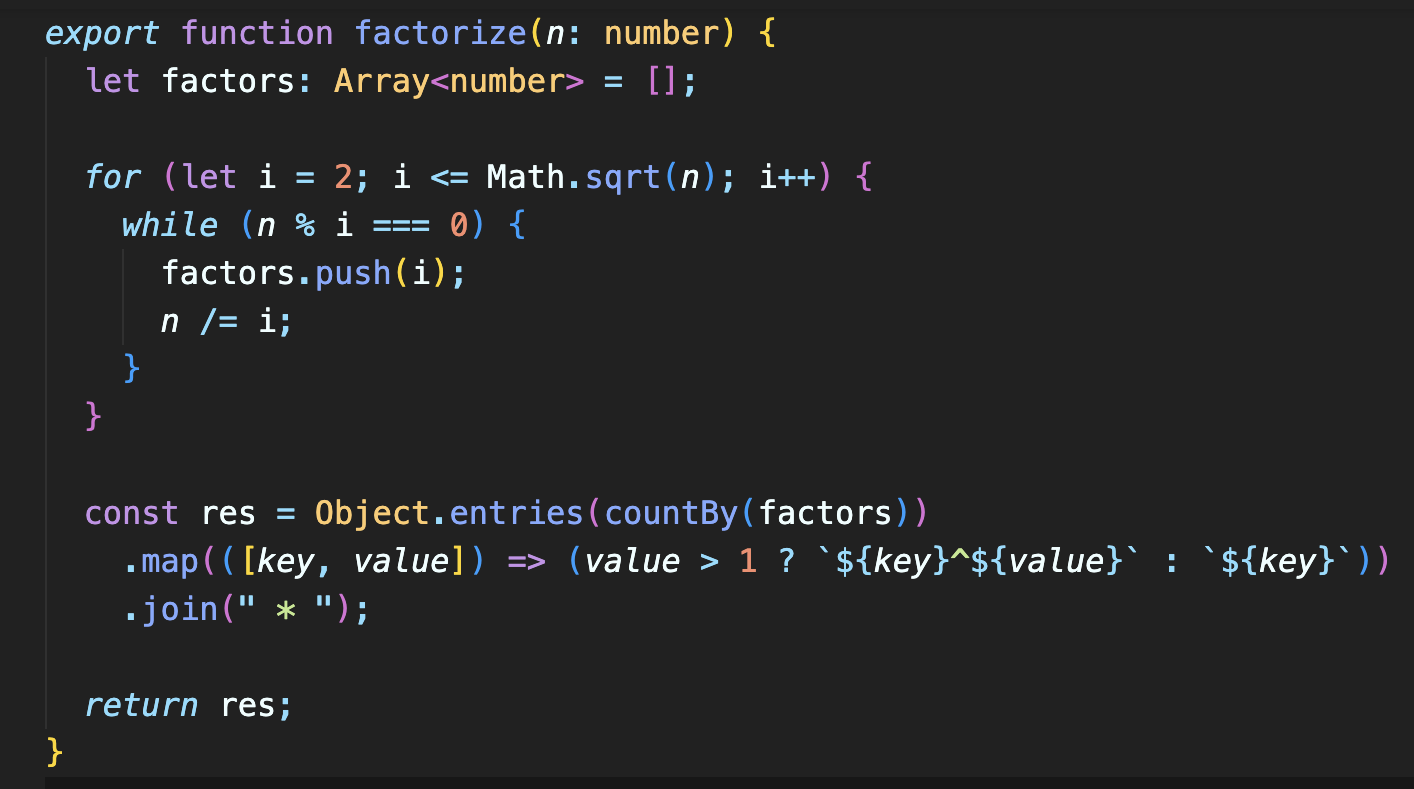


Рисунок 3.1 – Функция представления числа в канонической форме

Данная функция принимает в качестве параметра некоторое число *n* и последовательно делит его на числа от 2 до *√n*. Эти числа записываются в список, который возвращает данная функция. По данному списку в дальнейшем можно циклически пройти и вывести все числа из канонического разложения. Вывод данной функции представлен на рисунке 3.2.

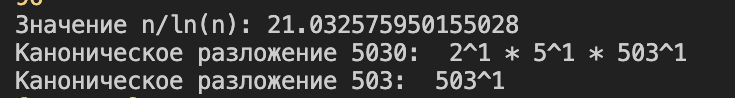


Рисунок 3.2 – Вывод канонической формы числа

# Проверка числа на простоту

Для проверки числа на простоту можно использовать следующий алгоритм: в цикле чисел от 2 до *√n* проверить, делится ли исходное число на одно из них. Если исходное число не делится ни на одно в заданном диапазоне, то оно является простым. Реализация функции проверки числа на простоту представлена на рисунке 4.1.

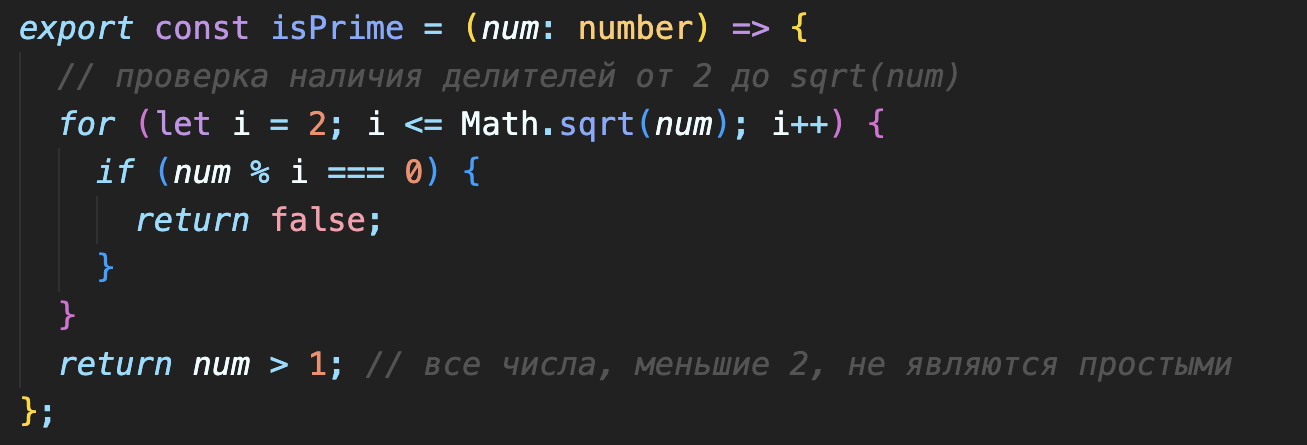


Рисунок 4.1 – Функция проверки числа на простоту

Далее, необходимо вызвать метод проверки числа на простоту, передав в параметры возвращаемое значение функции конкатенации чисел. Вывод данной функции для чисел *m* = 421 и *n* = 457 представлен на рисунке 4.3.

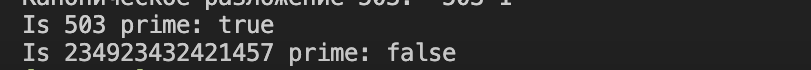


Рисунок 4.3 – Вывод функции проверки на простоту конкатенации чисел

**Вывод:** в данной лабораторной работе были приобретены практические навыки выполнения операций с числами для решения задач в области криптографии, и разработано приложение для автоматизации этих операций.