МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждения образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра программной инженерии

Специальность 1-40 01 01 Программное обеспечение информационных технологий

Направление специальности 1-40 01 01 10 Программное обеспечение информационных технологий (программирование интернет-приложений)

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ:

по дисциплине «Информационная безопасность»

Исполнитель

студент (ка) 3 курса группы 6 Розель Станислав Александрович

(Ф.И.О.)

Минск 2024

**1. Используя L\_PROST, найти все простые числа в интервале [2, n].**

Все простые числа в промежутке [2, 591] представлены на рисунке 1.1:

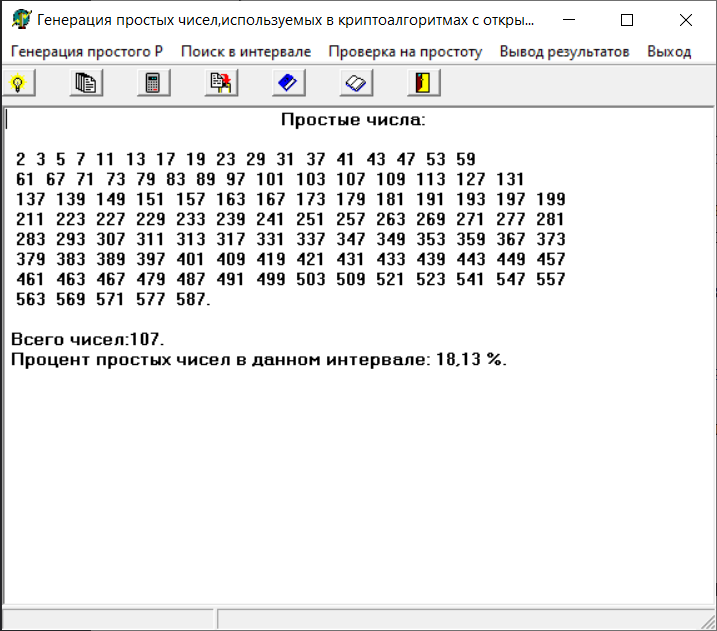


Рисунок 1.1 – Простые числа

Простых чисел: 107,

= 92,34

**2. Повторить п. 1 для интервала [m, n].**

Все простые числа в промежутке [555, 591] представлены на рисунке 2.1:

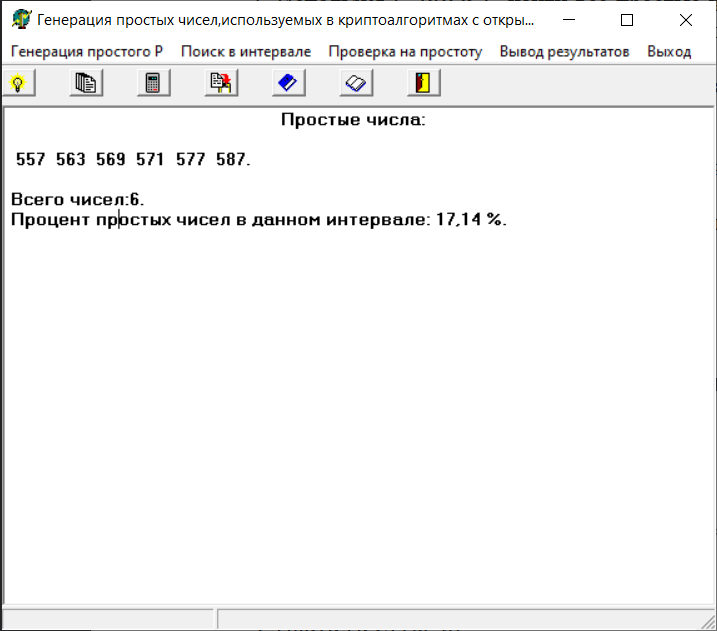


Рисунок 2.1 – Простые числа

Простых чисел: 6

= 10,04

Таким образом простыми числами в диапазоне [555, 591]: **557, 563, 569, 571, 577, 587**

**3. Записать числа m и n в виде произведения простых множителей (форма записи – каноническая).**

555: 3\*5\*37

591: 3\*197

**4. Проверить, является ли число, состоящее из конкатенации цифр m ǀǀ n (табл. 1.2), простым.**

= 745,37

5 + 5 + 5 + 5 + 9 + 1 = 30, сумма цифр дающих число кратное трем, является кратным трем

555 591 не является простым

**5. Найти НОД (m, n)**

НОД(555,591)=3

**6. Разработать авторское приложение в соответствии с целью лабораторной работы.**

В листинге 6.1 представлен код авторского приложения

|  |
| --- |
| const readline = require("readline");  const rl = readline.createInterface({  input: process.stdin,  output: process.stdout,  });  function isPrime(num) {  if (num < 2) return false;  for (let i = 2; i <= Math.sqrt(num); i++) {  if (num % i === 0) return false;  }  return true;  }  function findPrimesInRange(start, end) {  const primes = [];  for (let i = start; i <= end; i++) {  if (isPrime(i)) primes.push(i);  }  return primes;  }  function gcdTwoNumbers(a, b) {  while (b !== 0) {  const temp = b;  b = a % b;  a = temp;  }  return a;  }  function gcdMultipleNumbers(numbers) {  return numbers.reduce((acc, num) => gcdTwoNumbers(acc, num));  }  function showMenu() {  console.log("\nВыберите действие:");  console.log("1. Найти простые числа в диапазоне");  console.log("2. Найти НОД для нескольких чисел");  console.log("3. Выйти");  rl.question("Ваш выбор: ", handleMenu);  }  function handleMenu(choice) {  switch (choice.trim()) {  case "1":  askForRange();  break;  case "2":  askForNumbers();  break;  case "3":  console.log("Выход из программы. До свидания!");  rl.close();  break;  default:  console.log("Некорректный выбор. Попробуйте снова.");  showMenu();  break;  }  }  function askForRange() {  rl.question("Введите начало диапазона: ", (startInput) => {  rl.question("Введите конец диапазона: ", (endInput) => {  const start = parseInt(startInput);  const end = parseInt(endInput);  if (isNaN(start) || isNaN(end)) {  console.log("Оба значения должны быть числами. Попробуйте снова.");  askForRange();  } else if (start > end) {  console.log("Начало диапазона не может быть больше конца. Попробуйте снова.");  askForRange();  } else {  const primes = findPrimesInRange(start, end);  console.log(`Простые числа в диапазоне от ${start} до ${end}: ${primes.join(", ")}`);  showMenu();  }  });  });  }  function askForNumbers() {  rl.question("Введите числа через пробел (не более 3 чисел): ", (input) => {  const numbers = input  .split(" ")  .map(Number)  .filter((num) => !isNaN(num));  switch (numbers.length) {  case 0:  console.log("Вы не ввели ни одного числа. Попробуйте снова.");  askForNumbers();  break;  case 1:  console.log(  `Вы ввели одно число: ${numbers[0]}. НОД для одного числа — это само число. Попробуйте снова.`  );  askForNumbers();  break;  case 2:  case 3:  const gcd = gcdMultipleNumbers(numbers);  console.log(`НОД для чисел ${numbers.join(", ")}: ${gcd}`);  showMenu();  break;    default:  console.log("Вы ввели больше трёх чисел. Попробуйте снова.");  askForNumbers();  break;  }  });  }  console.log("Добро пожаловать в программу!");  showMenu(); |

Листинг 6.1 – Код приложения

**7. С помощью созданного приложения выполнить задания по условиям п. 1 и 2.**

На рисунке 7.1 представлен результат работы

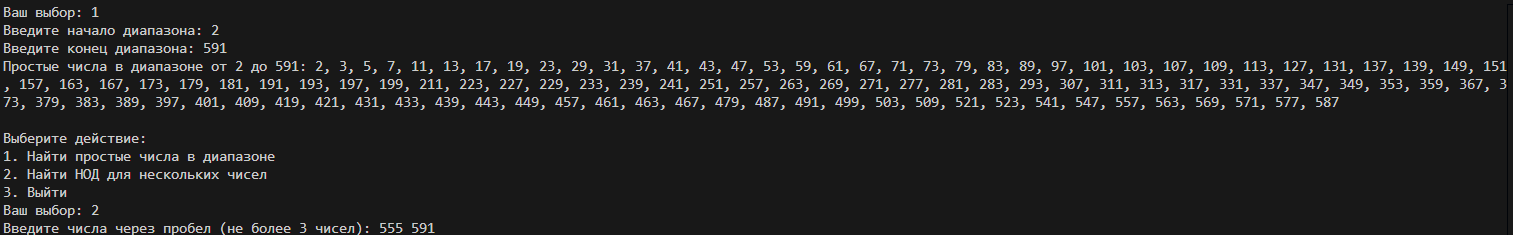


Рисунок 7.1 – Результат работы