Основные требования

- В процессе выполнения практических работ №1 - №7 настоятельно рекомендуется использовать один выбранный язык программирования.
- В процессе выполнения практических работ №1 - №7 НЕ допускается использование готовых библиотечных решений за исключением упомянутых в «основных требованиях».
- К концу курса на основе выполненных практических работ №1 - №7 необходимо собрать библиотеку для генерации больших простых чисел, которые используются в алгоритмах из лабораторных работ.
- Для проверки алгоритмов из практических работ №2 - №5 рекомендуется написать дополнительную программу с использованием библиотеки GMP для генерации больших чисел.

Практическая работа №1 «Большие числа»

1. Написать реализацию класса больших чисел.
2. Написать программу для реализации арифметических операций с большими числами: сложение, вычитание, деление, умножение и возведение в степень.
3. Добавить в программу из пункта №2 функционал сравнение двух больших чисел, поиска их наименьшего общего кратного (НОК) и наибольшего общего делителя (НОД).

Практическая работа №2 «Большие простые числа и числа Мерсенна»

1. С помощью программы из практической работы №1 вычислить значение $(2^{136279841} – 1)$ — самого большого из посчитанных простых чисел. Измерить скорость вычисления.
2. Проверить число из пункта №1 на простоту с помощью «стандартного» метода проверки на простоту:
    - проверка на чётность,
    - проверка на деление на 5,
    - проверка на деление суммы цифр на 3, 6 и 9,
    - проверка всех чисел до квадратного корня.

    Написать для этого собственную реализацию метода, используя в качестве основы практическую работу №1.
3. Проверить число из пункта №1 на простоту с помощью «Решета Эратосфена». Написать для этого собственную реализацию метода, используя в качестве основы практическую работу №1.
4. Проверить число из пункта №1 на простоту с помощью «Решета Аткина». Написать для этого собственную реализацию метода, используя в качестве основы практическую работу №1.
5. Проверить число из пункта №1 на простоту с помощью теста Люка-Лемера. Написать для этого собственную реализацию метода, используя в качестве основы практическую работу №1.

Практическая работа №3 «Вероятностные методы проверки на простоту»

1. Написать реализацию вероятностного теста Миллера-Рабина. Проверить известное простое число 100 раз, собрать статистику.
2. Написать реализацию вероятностного теста Люка на сильную псевдопростоту. Проверить известное простое число 100 раз, собрать статистику.
3. Написать реализацию вероятностного теста Бейли-Померанца-Селфриджа-Уогстаффа (обобщение тестов из пункта №1 и пункта №2). Проверить известное простое число 100 раз, собрать статистику.
4. Сравнить результаты проверки одного и того же простого числа с помощью тестов из пунктов №1 - №3.

Практическая работа №4 «Детерминированные методы проверки на простоту»

1. Написать реализацию детерминированного теста Агравала-Каяла-Саксены. Проверить известное простое число ($< 2^{32}$), измерить скорость вычислений. Проверить известное простое число ($>2^{64}$), измерить скорость вычислений. Сравнить результаты.
2. Написать реализацию детерминированного теста Миллера. Проверить известное простое число ($< 2^{32}$), измерить скорость вычислений. Проверить известное простое число ($>2^{64}$), измерить скорость вычислений. Сравнить результаты.
3. Сравнить результаты проверки одного и того же простого числа с помощью тестов из пунктов №1 - №2.

Практическая работа №5 «Универсальный метод проверки на простоту»

1. Написать реализацию универсального алгоритма Аткина-Морейна на эллиптических кривых (ECPP). Проверить известное простое число ($< 2^{32}$), измерить скорость вычислений. Проверить известное простое число ($>2^{64}$), измерить скорость вычислений.
2. Сравнить результаты из пункта №1 с результатами, полученными в практической работе №4.
3. Проверить на простоту числа ($2^{82589933} - 1$) и ($2^{136279841} - 1$), измерить скорость вычислений.

Практическая работа №6 «Случайные числа»

1. Написать линейный конгруэнтный генератор псевдослучайных чисел. Сгенерировать 100, 1000, 10000 значений, определить математическое ожидание, дисперсию, количество совпадающих значений. Построить графики.
2. Написать программу для генерации случайных чисел методом вихря Мерсенна. Сгенерировать 100, 1000, 10000 значений, определить математическое ожидание, дисперсию, количество совпадающих значений. Сравнить значения характеристик с полученными в пункте №1.
3. Написать программу для генерации случайных чисел с использованием накопления энтропии.

Возможные варианты:

Вариант	Источник энтропии
1	Аудиозапись (амплитуда, частота)
2	Изображение (пиксели)
3	Посимвольная печать заданной строки (подсчёт времени между нажатиями клавиш)
4	Положение курсора на экране
5	Время прихода прерывания от внешнего устройства (мышь, клавиатура)
6	Частоты появления заданных символов в тексте

Допустимо использование альтернативных источников энтропии.

Практическая работа №7 «Случайные простые числа»

1. Написать генератор случайных простых чисел размером 512 бит, 512 байт, 1 Кбайт, 32 Кбайта, 128 Кбайт, 256 Кбайт, 512 Кбайт, 1 Мбайт. Размер числа должен вводиться с клавиатуры. Допустимы другие значения размера, не превышающие 1 Мбайт. В качестве генератора начальных значений использовать программу из пункта №3 практической работы №6, а в качестве алгоритма проверки на простоту — алгоритм ECPP.
2. Сгенерировать 1000 простых чисел размером 512 бит, 512 байт, 1 Кбайт и 32 Кбайта. Для каждой генерации среди полученных чисел найти совпадающие, посчитать их количество, вычислить математическое ожидание и дисперсию.
3. Проверить генератор случайных простых чисел путём генерации числа размером 32 Мбайта. Измерить скорость работы.