**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский Авиационный Институт»**

**(Национальный Исследовательский Университет)**

**Институт: №8 «Информационные технологии   
и прикладная математика»   
Кафедра: 806 «Вычислительная математика   
и программирование»**

Лабораторная работа № 2   
по курсу «Криптография»

Группа: М8О-307Б-21

Студент: Дубровин Д.К.

Преподаватель: А. В. Борисов

Оценка:

Дата: 10.09.2024

Москва, 2024

**ОГЛАВЛЕНИЕ**

[1 Тема 3](#_Toc158983147)

[2 Задание 3](#_Toc158983148)

[3 Теория 4](#_Toc158983149)

[4 Ход лабораторной работы 5](#_Toc158983150)

[5 Выводы 7](#_Toc158983151)

# **Тема**

Факторизация чисел

# **Задание**

Разложить каждое из чисел и на нетривиальные сомножители. Вариант №7:

# **Теория**

Факторизация числа — это процесс разделения числа на несколько множителей так, что при их перемножении получается исходное число. Если говорить простыми словами, это похоже на разбиение числа на кусочки, которые являются меньшими числами, и все эти кусочки при умножении друг на друга дают первоначальное число. Обычно, когда мы говорим о факторизации, мы имеем в виду разбиение числа на простые множители, то есть такие числа, которые делятся только на 1 и на самих себя. Например, факторизация числа 12 будет 2 × 2 × 3, где 2 и 3 являются простыми числами.

# **Ход лабораторной работы**

При просмотре задания у меня закрались подозрения, что тут есть подвох. Я узнал в интернете что есть разные методы для факторизации числа. Мне больше приглянулся Ро-алгоритм Полларда. Ну я изучил, реализовал на языке Python и тут появился нюанс, что считать то он будет долго. Очень долго. Я уже просрочу срок сдачи лабораторной. Я посмотрел на другие алгоритмы в надежде найти самый быстрый. Но как-то все сомнительно выглядело. Да и самый быстрый алгоритм реализовать как-то сложновато. Я даже посмотрел в книжке под названием «Алгебраические основы криптографии» от Э.А. Применко. Но там ничего не нашел, хотя я не знаю, что я должен был там найти.

Я решил узнать у своей подруги со старших курсов в чем суть. Мне поведали, что, в общем и целом, я не смогу ее сделать на основе своих ресурсов. Поэтому есть опция прибегнуть к готовым решениям. Это оказались библиотека `mseive` на языке C (метод решета числового поля) и сайт <https://www.cryptool.org/en/cto/msieve>. Я прибегнул ко второму способу и получил такой результат:

1. 260951289862485772644727258162652873363
2. 416791782672403295662841737728685758229

Со вторым числом мне дали наводку в какую сторону копать. Суть заключается в том, что при факторизации второго числа первый множитель находится как НОД с числом другого варианта, а второй множитель – простым делением первоначального и первого множителя.

Вот что получилось:

1. 163397696065821074680902655996825570159706795236045906521559460962578519078856105725648968556569072711406616529723182939501812794722662366814883631619640072792920581850719503493330646427755230896373119814690571985811278115577251609542362580175148578313739080898244696381665260084479643389434792645421908712913
2. 9864065454751219846168673527138762130771017421052429595374272499175002527482210596174648387419880628453282971116194905653753945702097018977536037164679168

# **Выводы**

Первоначально я не понимал идеи лабораторной. Но когда я узнал, как делать, то я восхитился уловкой. Отдаю дань уважения преподавателю, который придумал это, и первому студенту, который додумался до этого. Это показало, что не нужно все задачи решать в тупую, нужно искать какие-то альтернативные способы, а не думать, что задача нерешаема. Так же было полезно узнать о том какие есть алгоритмы факторизации. Отличная лабораторная работа!

# **Список используемой литературы**

1. <https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D1%86%D0%B5%D0%BB%D1%8B%D1%85_%D1%87%D0%B8%D1%81%D0%B5%D0%BB>
2. <https://habr.com/ru/sandbox/163811/>
3. <https://algorithmica.org/ru/pollard>