Отчет лабораторной работы №1

По дисциплине “Криптография”

Выполнил Кондратьев Егор М8О-306Б-19

Вариант 14

(n1 – 77 чисел, n2 – 463 чисел)

n1=61121970174911146319545193754425119520875945215282784640177276523929376501913

n2=6238596931990131478275327152343801799668257762705582576427997815976220309207691211435205004903767290173230872141307213029392296401324769866348679434743265963571375902347394343714110590043962617817326709729518180348452284402707055765832648925000626213596053865173116238603592051986073329528908502125295911837124501151097345605082152827287279961240150766589609665614675527012229363982035766005443366925574863537569235988977028475559932462781742771732084527629722071

В олимппроге часто встречаются задачи по теории чисел. И кстати задача разложения числа на простые множители одна из самых частых. Поэтому я, конечно, знаю асимптотику алгоритмов для решения таких задач и это в основном: O(n \* log n) и O(sqrt n). Их, конечно, достаточно для решения контестов, но для выполнения данной лабки они будут так себе…. мягко сказано….хех.

Поэтому я начал искать эффективные методы решения и нашел - msieve(факторизация больших чисел с помощью квадратичного решета).

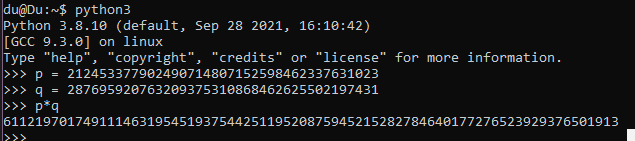
Мне не понравилось его устанавливать, так как это какие-то танцы с бубном получались, поэтому тупо нашел онлайн-инструмент: <https://www.cryptool.org/en/cto/msieve>

В итоге за 2 минуты, ответ на первое число был найден:

p = 212453377902490714807152598462337631023

q = 287695920763209375310868462625502197431

что прошло мою нехитрую проверку



Второе число сайт не смог считать из-за ограничения на длину. Даже если не учитывать ограничение на длину, сайт использовать всё равно нет смысла, так как алгоритм в среднем потратил бы более 20 часов на факторизацию второго числа…к чему я конечно не готов.

Здесь я понимаю, тут не обойдется без НОДов и чтобы не пытаться рандомить, можно воспользоваться числами n2 из других вариантов, а чтобы было больше шансов, можно проверить их все.

И, если найдется НОД, будем считать его первым нашим ответом q. А чтобы найти второе p, мы можем бесхитростно поделить наше n2 на НОД, и ТАДАМ. Но это пока теория, осталось реализовать. Что явно будет быстрым.

from math import gcd

num14 = 6238596931990131478275327152343801799668257762705582576427997815976220309207691211435205004903767290173230872141307213029392296401324769866348679434743265963571375902347394343714110590043962617817326709729518180348452284402707055765832648925000626213596053865173116238603592051986073329528908502125295911837124501151097345605082152827287279961240150766589609665614675527012229363982035766005443366925574863537569235988977028475559932462781742771732084527629722071

with open('n.txt') as f:

for line in f:

if line[0] != 'n':

var = line[0: -1]

else:

num = int(line[3: -2])

gcd\_n = gcd(num14, num)

if num == num14:

print("Одно и тоже")

elif gcd\_n != 1:

tmp = num14 // gcd\_n

print("Вариант {}, {} -- Сопряженное число, подошло".format(var, line[0: 2]))

print("n{} = {}\n".format(line[1], num))

print("q = {} (digits): {}\n".format(len(str(gcd\_n)), gcd\_n))

print("p = {} (digits): {}\n".format(len(str(tmp)), tmp))

check = tmp \* gcd\_n

print("Нехитрая проверка\nq \* p = {}".format(check))

exit()

else:

print("Вариант {}, {} : q = {} - печально,это нам не подходит(".format(var, line[0: 2], gcd\_n))

C:/Users/egork/Desktop/space/crypto/lab1/main.py

Вариант 0, n1 : q = 1 - печально, это нам не подходит(

Вариант 0, n2 : q = 1 - печально, это нам не подходит(

Вариант 1, n1 : q = 1 - печально, это нам не подходит(

Вариант 1, n2 : q = 1 - печально, это нам не подходит(

Вариант 2, n1 : q = 1 - печально, это нам не подходит(

Вариант 2, n2 : q = 1 - печально, это нам не подходит(

Вариант 3, n1 : q = 1 - печально, это нам не подходит(

Вариант 3, n2 : q = 1 - печально, это нам не подходит(

Вариант 4, n1 : q = 1 - печально, это нам не подходит(

Вариант 4, n2 : q = 1 - печально, это нам не подходит(

Вариант 5, n1 : q = 1 - печально, это нам не подходит(

Вариант 5, n2 : q = 1 - печально, это нам не подходит(

Вариант 6, n1 : q = 1 - печально, это нам не подходит(

Вариант 6, n2 : q = 1 - печально, это нам не подходит(

Вариант 7, n1 : q = 1 - печально, это нам не подходит(

Вариант 7, n2 : q = 1 - печально, это нам не подходит(

Вариант 8, n1 : q = 1 - печально, это нам не подходит(

Вариант 8, n2 : q = 1 - печально, это нам не подходит(

Вариант 9, n1 : q = 1 - печально, это нам не подходит(

Вариант 9, n2 : q = 1 - печально, это нам не подходит(

Вариант 10, n1 : q = 1 - печально, это нам не подходит(

Вариант 10, n2 : q = 1 - печально, это нам не подходит(

Вариант 11, n1 : q = 1 - печально, это нам не подходит(

Вариант 11, n2 : q = 1 - печально, это нам не подходит(

Вариант 12, n1 : q = 1 - печально, это нам не подходит(

Вариант 12, n2 : q = 1 - печально, это нам не подходит(

Вариант 13, n1 : q = 1 - печально, это нам не подходит(

Вариант 13, n2 : q = 1 - печально, это нам не подходит(

Вариант 14, n1 : q = 1 - печально, это нам не подходит(

Одно и тоже

Вариант 15, n1 : q = 1 - печально, это нам не подходит(

Вариант 15, n2 -- Сопряженное число, подошло

n2 = 9173108187535281517140762116700384612326624554619159756170327131075765663592428184342498108166844987875472192379425402617261537751017461789175811281066290112144995485771928277678645046268515605836413638915408097220188140275180089310734305255138886443749996612234117011911904572687273790898184986984786012309336821986211719508683069979735549322015707035016397961277189356172028200502143324154428183926213506337495841035478668065439542819480000104949666864308342553

q = 155 (digits): 26028460724297046804451005423500883088198628651710108650506896689755792570919576728631986295804878634384681797646296297818609601926376801836356334018530619

p = 309 (digits): 239683667738619909435043761168696430649299798361354713695469181019038516148277746444115246915396350327452542286471681408878360451028459196986757532118505371367962676637351003668238330569849547710157828613626353510304160905306882076194976453001152478362196953870757049251760897968794570268428701318813938623509

Нехитрая проверка

q \* p = 6238596931990131478275327152343801799668257762705582576427997815976220309207691211435205004903767290173230872141307213029392296401324769866348679434743265963571375902347394343714110590043962617817326709729518180348452284402707055765832648925000626213596053865173116238603592051986073329528908502125295911837124501151097345605082152827287279961240150766589609665614675527012229363982035766005443366925574863537569235988977028475559932462781742771732084527629722071

Process finished with exit code 0

И да, получилось, странный способ, конечно, но работает, а значит трогать не надо.

в итоге получились такие ответы:

**N1:**

1. p = 212453377902490714807152598462337631023
2. q = 287695920763209375310868462625502197431

**N2:**

1. q = 155 (digits): 26028460724297046804451005423500883088198628651710108650506896689755792570919576728631986295804878634384681797646296297818609601926376801836356334018530619
2. p = 309 (digits): 239683667738619909435043761168696430649299798361354713695469181019038516148277746444115246915396350327452542286471681408878360451028459196986757532118505371367962676637351003668238330569849547710157828613626353510304160905306882076194976453001152478362196953870757049251760897968794570268428701318813938623509