Прагматика выполнения лабораторной работы

Студенты должны разбираться в методах шифрования. Поэтому освоение и реализации нахождения простых множителей разнообразными методами положительно скажется на будущее понимание процесса шифрования.

Цель выполнения лабораторной работы

Реализовать с помощью программирования метод поиска простых множителей, методами, описанными в задании к лабораторной работе №6.

Задачи выполнения лабораторной работы

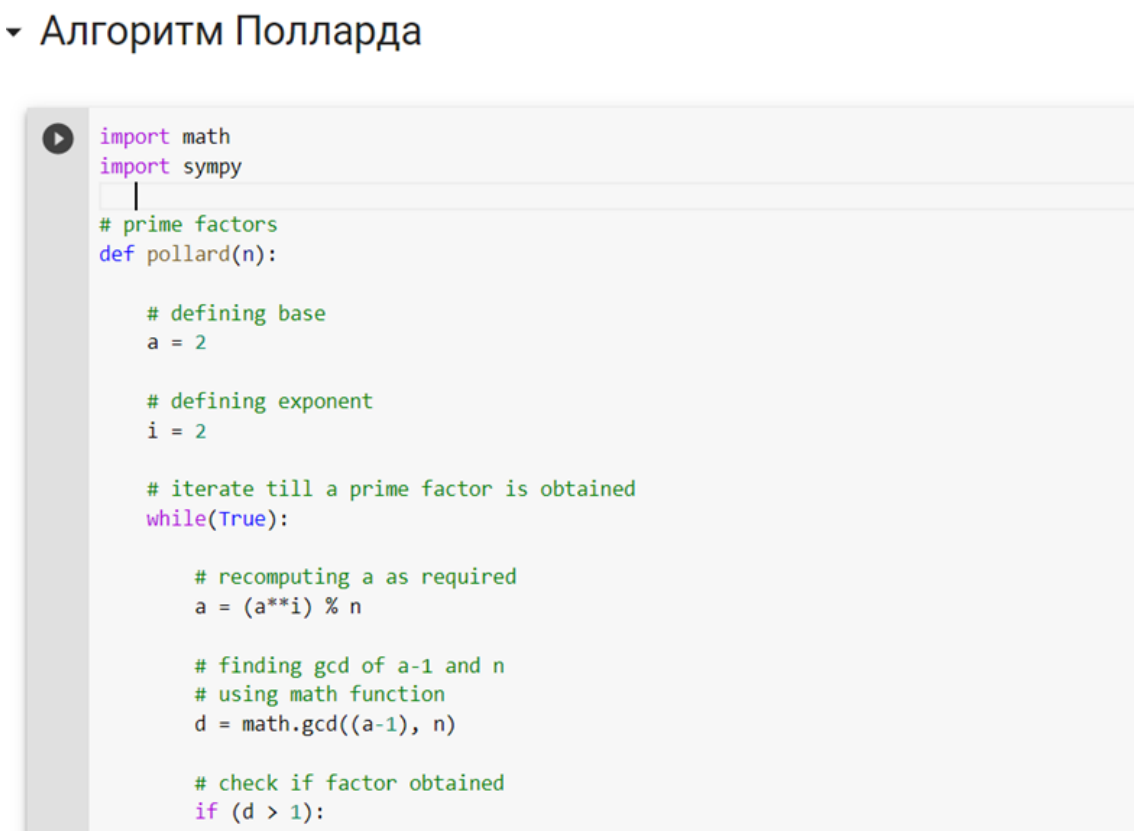
Разработать код программы, который будут представлять из себя:

1. Алгоритм Полларда.

Результаты выполнения лабораторной работы

1. Реализовала программу по Алгоритму Полларда.

   (рис. -@fig:001)



   Основная суть алгоритма лежит в том, чтобы найти для числа n разложение на простые множители. Для работы данного алгоритма, число n должно быть обязательно нечетное, иначе мы получим зацикливание. Алгоритм Полларда представляет из себя нахождение не всех простых множителей, а только нетривиальных (всех, кроме 1 и самого n)/

Результаты выполнения лабораторной работы

2. Опробовала программный код на числе 1032225513.

(рис. -@fig:003)

```
while(True):

    # function call
    d = pollard(num)

    # add obtained factor to list
    ans.append(d)

    # reduce n
    r = int(num/d)

    # check for prime using sympy
    if(sympy.isprime(r)):

        # both prime factors obtained
        ans.append(r)

        break

    # reduced n is not prime, so repeat
    else:

        num = r

# print the result
print("Prime factors of", n, "are", *ans)
```

```
Prime factors of 1032225513 are 3 11 3079 10159
```

Я хотела ввести номер своего студенческого, но, к сожалению, он четный((. Поэтому я выбрала число, близкое к нему. По итогу выполнений цикла (проверки всех цифр от 2 до n-1), программа выдала список чисел. Их перемножение снова дает нам мое первоначальное число.

Вывод

Освоила на практике написание алгоритмов разложения на простые числа.

{.standout}Спасибо за внимание