Отчет по лабораторной работе №1

«Основные операторы и конструкции Python.»

Вариант 1. Кролики и рекуррентные отношения

**Цель работы:** Разработать программу на языке Python, моделирующую рост популяции кроликов с учетом заданных параметров и рекуррентного отношения, и вычисляющую общее число пар кроликов на заданный месяц.

**Задачи:**

1. Изучить задачу о размножении кроликов, предложенную Фибоначчи, и понять рекуррентную зависимость, описывающую рост популяции.

2. Адаптировать рекуррентное отношение для модели популяции, в которой каждая половозрелая пара кроликов рожает k пар кроликов вместо одной.

3. Реализовать программу на Python, принимающую на вход номер месяца (n) и количество рождаемых пар кроликов (k) и возвращающую общее число пар кроликов на n-ый месяц.

4. Протестировать программу с использованием примера входных данных и убедиться в правильности работы.

**Инструменты:**

\* **Язык программирования:** Python 3

\* **Среда разработки:** VS Code

\* **Алгоритм:**

1. Создание двух переменных F1 и F2, представляющих собой количество пар кроликов на первый и второй месяцы (обе равны 1).

2. Последовательное выполнение действий для месяцев с третьего по n-ый. В каждом цикле:

\* Вычисление количества пар кроликов на текущий месяц (F\_current) по формуле: F\_current = F2 + k \* F1, где F2 - количество пар кроликов на предыдущий месяц, F1 - количество пар кроликов за два месяца до текущего, а k - количество пар кроликов, рождаемых каждой половозрелой парой.

\* Обновление переменных F1 и F2 для следующей итерации: F1 = F2, F2 = F\_current.

3. После завершения цикла возвращается значение F2, которое содержит общее число пар кроликов на n-ый месяц.

**Код программы (Python):**

def fibonacci\_rabbits (n, k):

# Инициализация начальных условий

F1, F2 = 1, 1

# Расчет количества пар кроликов на n-й месяц

for i in range(3, n + 1):

F\_current = F2 + k \* F1

F1 = F2

F2 = F\_current

return F2

print(fibonacci\_rabbits (5, 3))

**Ошибки и их исправления:**

1. **Ошибка:** Изначально в коде была допущена ошибка в цикле. Цикл начинался с range(2, n + 1), что приводило к неправильному расчету количества пар кроликов, начиная с третьего месяца. Фактически, в программе просто не происходило вычислений.

• **Исправление:** Исправлено начало цикла на range(3, n + 1), что позволило правильно последовательно выполнять действия для перебора месяцев и применять рекуррентное отношение.

**Выводы:**

В результате выполнения лабораторной работы была разработана программа на языке Python, моделирующая рост популяции кроликов с учетом коэффициента размножения k. Программа эффективно вычисляет общее число пар кроликов на заданный месяц n, используя рекуррентное отношение.

Работа позволила закрепить знания об основных операторах и конструкциях языка Python, а также продемонстрировала применение рекуррентных отношений для моделирования динамических процессов. Использование динамического программирования позволило эффективно решить задачу, вычисляя результаты, опираясь на решения для более простых случаев.