

Отчёт

Практическое занятие №4

Тема: Составление программ циклической структуры в IDE VS Code.

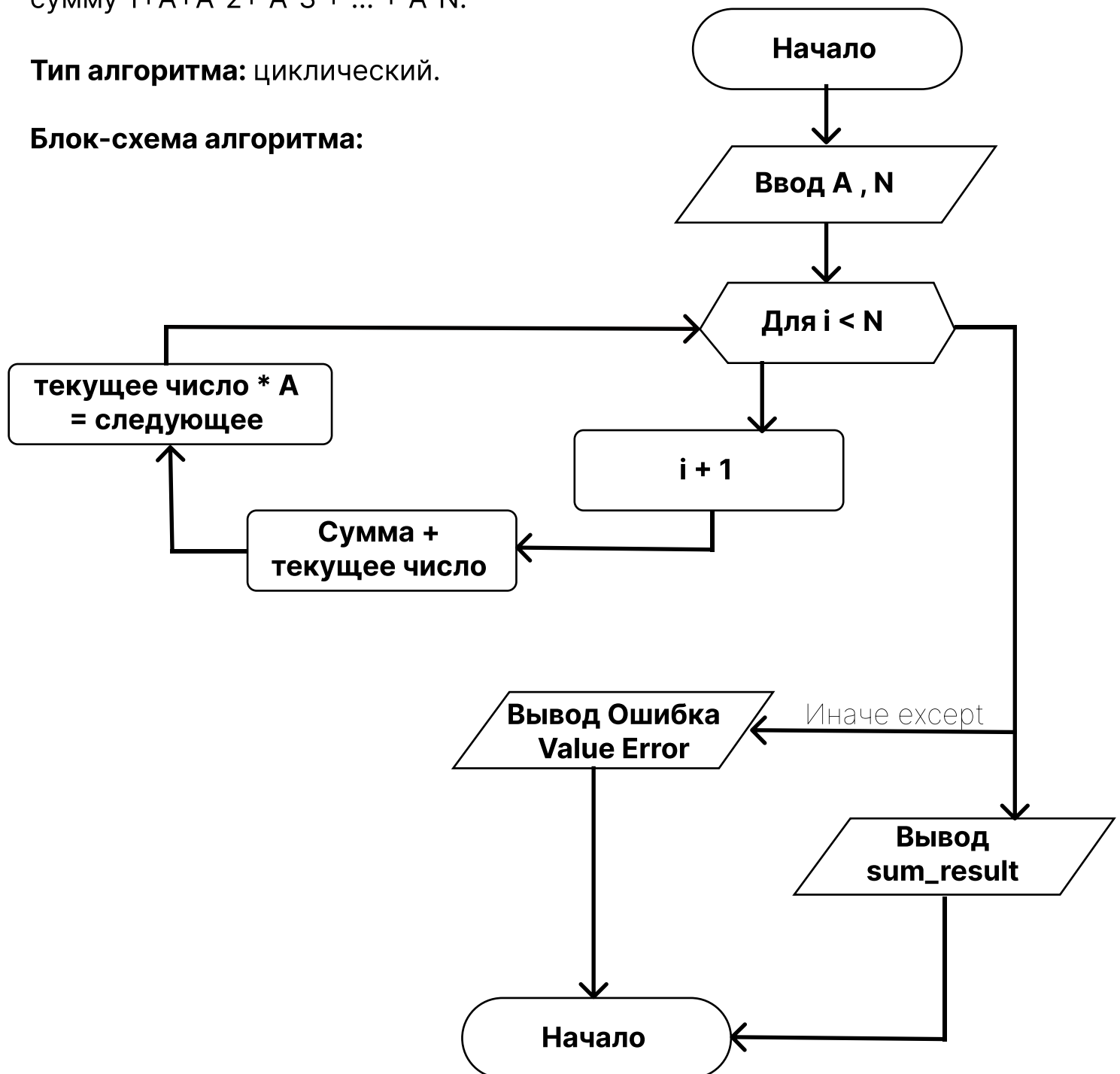
Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ циклической структуры в IDE VS Code.

Постановка задачи.

Дано вещественное число A и целое число N (>0). Используя один цикл, найти сумму $1+A+A^2+A^3+\dots+A^N$.

Тип алгоритма: циклический.

Блок-схема алгоритма:



Текст программы:

```

#Дано вещественное число A и целое число N (>0). Используя один цикл, найти сумму
1+A+A^2+A^3+...+A^N.
try:
    # Вводим вещественное число A
    A = float(input("Введите вещественное число A: "))

    # Вводим целое число N, которое должно быть больше 0
    N = int(input("Введите целое число N (>0): "))

    # Проверяем, что N больше 0, иначе вызовем исключение ValueError
    if N <= 0:
        raise ValueError("Число N должно быть больше 0.")

    # Инициализируем переменную для хранения суммы
    sum_result = 0

    # Инициализируем переменную для текущего члена последовательности, начиная с A^0 = 1
    current_term = 1

    # Используем цикл для вычисления суммы степеней от A^0 до A^N
    for i in range(N + 1): # Проходим от 0 до N включительно
        sum_result += current_term # Добавляем текущий член к сумме
        current_term *= A # Переходим к следующему члену, умножая на A

    # Выводим итоговую сумму
    print("Сумма:", sum_result)

# Обработываем исключение ValueError для некорректных данных (например, если N <= 0)
except ValueError as e:
    print("Ошибка:", e)

```

Протокол работы программы:

Введите вещественное число A: 2
Введите целое число N (>0): 16
Сумма: 131071.0

Вывод: в процессе выполнения практического занятия выработал навыки составления программ циклической структуры в IDE Visual Studio: Code. Был использован метод **for**.

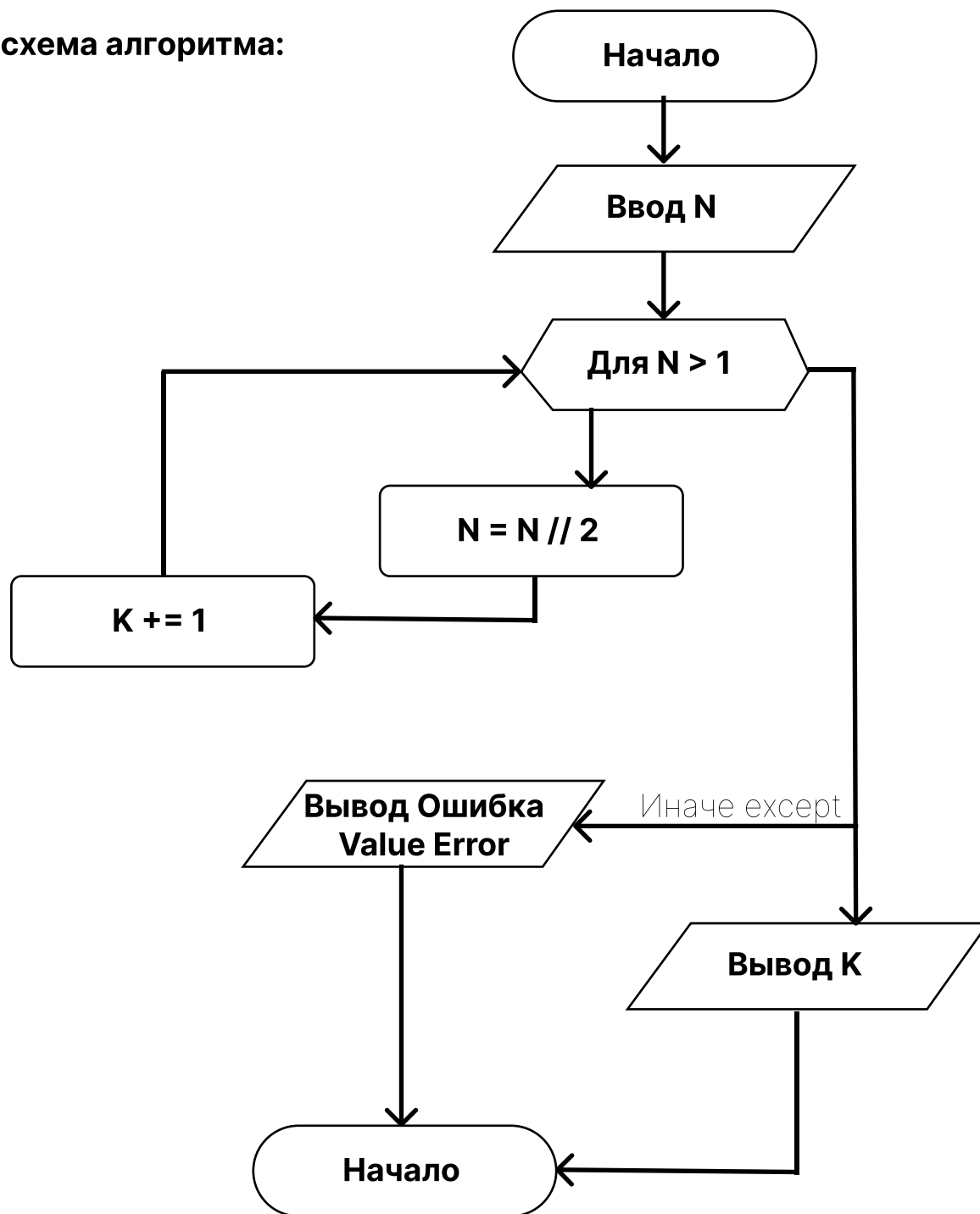
Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.

Постановка задачи.

Дано целое число N (>0), являющееся некоторой степенью числа 2: $N = 2^K$
Найти целое число K — показатель этой степени.

Тип алгоритма: циклический.

Блок-схема алгоритма:



Текст программы:

#Дано целое число $N (>0)$, являющееся некоторой степенью числа 2: $N = 2^K$. Найти целое число K - показатель этой степени.

try:

Вводим целое число N

$N = \text{int}(\text{input}(\text{"Введите целое число } N (>0), \text{ являющееся степенью числа 2: "}))$

Проверяем, что N является положительным и степенью числа 2

if $N \leq 0$ or $(N \& (N - 1)) \neq 0$:

raise ValueError("Число N должно быть положительным и степенью числа 2.")

Инициализируем переменную K для подсчета показателя степени

$K = 0$

Пока N больше 1, делим его на 2 и увеличиваем K на 1

while $N > 1$:

$N //= 2$ # Целочисленное деление на 2, уменьшаем N

$K += 1$ # Увеличиваем показатель степени

Выводим значение показателя степени K

print("Показатель степени K :", K)

Обработываем исключение ValueError для некорректных данных

except ValueError as e:

print("Ошибка:", e)

Протокол работы программы:

Введите целое число $N (>0)$, являющееся степенью числа 2: 128

Показатель степени K : 7

Вывод: в процессе выполнения практического занятия выработал навыки составления программ циклической структуры в IDE Visual Studio: Code. Был использован метод **for**.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.