Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Санкт-петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича»

(СПбГУТ)

ОтчЁт  
ЛАБОРАТОРНАЯ РАБОТА №4

Циклические вычислительные процессы. Вычисления по рекуррентным формулам

Руководитель,  
старший преподаватель Помогалова А. В.

подпись, дата

Исполнитель,  
группа ИКПИ-33 Коньков М. Д.

подпись, дата

УСЛОВИЯ РАБОТЫ

Вариант № 12:

Создать программу по вычислению значений функций y и z:

|  |  |  |  |
| --- | --- | --- | --- |
| Номер варианта | Функция | Рабочий набор | |
|  |  | *n* | *х* |
| 12 |  | *15* | *0,4* |

Общая формулировка:

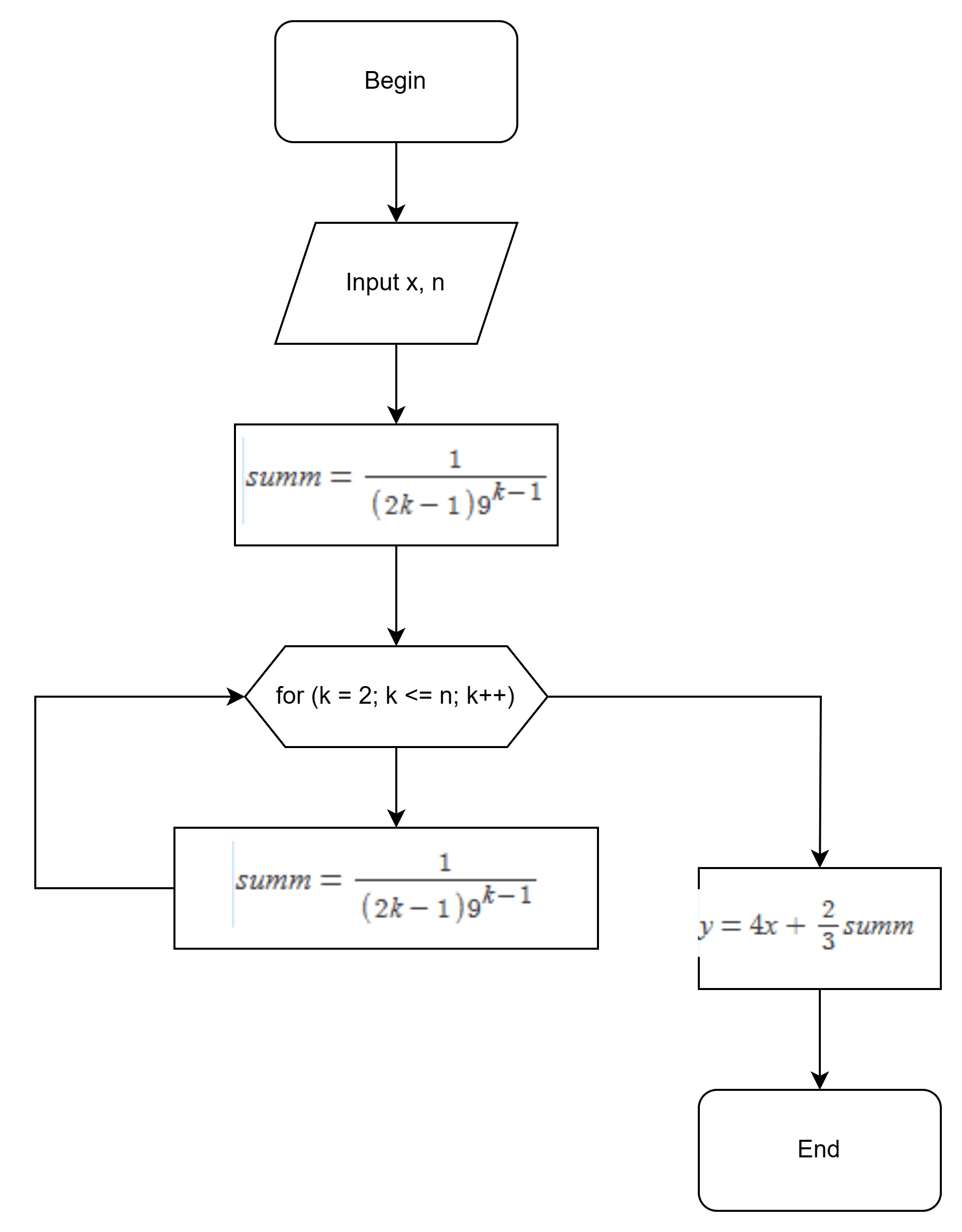
Решить задачу вычисления значений рекуррентной функции, содержащей сумму или (и) произведение.

# Общие алгоритм решения

Решаемая задача относится к категории задач формульного счета. Для нахождения переменной результата y необходимо пройти несколько итераций операции суммы и получить ответ. Это называется рекуррентной функцией.

В программе должен быть предусмотрен ввод исходных данных, к которым относятся переменные *x, n*; вычисления величин *y, summ*; вывод результатов вычислений (вывод значений величин *y* и *summ*).

## Общий алгоритм решения



|  |  |  |  |
| --- | --- | --- | --- |
| N | Обозначение в задаче | Идентификатор | Назначение |
| 1 | *y* | *y* | Результат |
| 2 | *x* | *x* | Исходные данные |
| 3 | *∑* | *summ* | Промежуточная величина |
| 4 | *n* | *n* | Исходные данные |
| 5 | *count* | *count* |

## Тестирование

Для тестирования программы выбираем контрольный набор исходных данных: *x* = 0.4, n = 15

|  |  |  |  |
| --- | --- | --- | --- |
| Переменная | x | n | y |
| Рабочий набор | 0.4 | 15 | 1.626481 |
| Контрольный набор | 0.5 | 10 | 2.026480 |

1. Листинг программы laba4.c (C)

#include <stdio.h>

#include <math.h>

int main(){

    int n, k = 1, i, again, count = 1;

    float x, y, summ;

    while(1){

printf("\033[0d\033[2J");

        // Entering variables and checking

        printf("Enter x: \n");

        scanf("%f", &x);

        printf("Enter n (more than 1): \n");

        scanf("%d", &n);

        if (n < 2){

            printf("Invalid value. Please, try again \n");

            continue;

        }

        summ = 1 / ((2 \* k - 1) \* pow(9, k - 1));

        printf("Result of summ: \n");

        printf("%d. %f\n", count, summ);

        for (k = 2; k <= n; k++){

            summ = summ + ( 1 / ((2 \* k - 1) \* pow(9, k-1)) );

            count += 1;

            printf("%d. %f\n", count, summ);

        }

        y = 4 \* x + (2.0 / 3.0) \* summ;

        printf("y = %f\n", y);

        // Checking for repeat

        printf("\nDo you want to try again? (1 - yes / 0 - no)\n");

        scanf("%d", &again);

        if (again == 1){

            continue;

        } else {

            return 0;

        }

    }

    return 0;

}

1. Результат работы программы laba4.c (С)

