УТВЕРЖДЕНО

Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ УПРАВЛЕНИЯ» Институт заочного образования

Разделяй и властвуй   
Загрузчик  
Рыбкин А.П.

|  |  |
| --- | --- |
| Подп. И дата |  |
| Инв.№ дубл. |  |
| Взам. Инв.№ |  |
| Подп. И дата |  |
| Инв.№ подп. |  |

2025

Литера

**1. Название программы**  
Разделяй и властвуй   
**2. Назначение программы**  
Программа предназначена для ввода пользователем массива чисел и вычисления их суммы с помощью рекурсивной функции.   
**3. Язык программирования**  
Python.   
**4. Описание алгоритма**   
Пользователь вводит последовательность чисел через пробел. - Числа преобразуются в список целых чисел. - Для подсчета суммы используется рекурсия: базовым случаем является пустой список, при этом возвращается 0. В противном случае берется первый элемент списка и добавляется к сумме остальных элементов, где остальные элементы обрабатываются рекурсивно.   
**5. Требования к входным данным**Входные данные должны представлять собой последовательность целых чисел, разделённых пробелом. - Все символы, кроме цифр и пробелов, игнорируются.

**6.Формат входных данных (пример):**

1 2 3 4 5

**7. Требования к выходным данным**Выходными данными должна быть сумма всех введённых чисел.

**8. Формат выходных данных (пример):**

Сумма введенных чисел: 15

**9. Ограничения**  
Программа поддерживает только целые числа. - Максимальная длина последовательности чисел ограничена возможностями стандартной библиотеки **Python. 10. Пример работы программы (пример):**

Введите числа через пробел: 1 2 3 4 5

Сумма введенных чисел: 15

**11. Дополнительная информация**  
Программа использует рекурсию для вычислений, что может быть полезно для понимания принципов рекурсивного подхода в программировании. Эта программа подходит для небольших последовательностей чисел. При работе с большими массивами рекомендуется использовать стандартные функции такие как sum(), так как они работают быстрее и эффективнее. Рекурсия нужна только в языках программирования таких как Haskell т.к в языках функционального программирования, циклов нет, поэтому для написания подобных функций приходится применять рекурсию.