Проблема: найти дубликат элемента в массиве

Задача: дан массив из n + 1 целого элемента между 1 и n. Необходимо найти дублирующийся элемент и вернуть его.

Пример:

Ввод: [1, 2, 2, 3]

Вывод: 2

Для решения этой задачи я буду использовать язык программирования C. Это самый быстрый высокоуровневый язык программирования, потому он будет крайне эффективен для данной задачи

Существует несколько вариантов поиска дублирующегося элемента. Рассмотрим некоторые из них:

1. Метод грубой силы

Этот метод заключается в последовательном переборе элементов и сравнением их с каждым следующим в массиве. Код на языке C имеет вид:

**int** brute\_force(**int** array[n + **1**], **int** n)

{

**for** (**int** i = **0**; i < n + **1**; i++)

{

**for** (**int** j = i + **1**; j < n + **1**; j++)

{

**if** (array[i] == array[j])

**return** array[i];

}

}

**return** -**1**;

}

Пожалуй, это самый простой алгоритм, однако он является самым затратным по времени. Его временная сложность составляет O(n2), что крайне неэффективно

1. Метод разницы сумм

В данном методе мы находим повторяющийся элементом путем вычисления разницы между всеми элементами массива и суммой ряда натуральных чисел от 1 до n. Код на языке C имеет вид:

**int** **sum**(**int** array[n + **1**], **int** n)

{

**int** temp = **0**;

**for** (**int** i = **0**; i < n + **1**; i++)

{

temp += array[i];

}

**return** temp - (n\*(n + **1**) / **2**);

}

Это интуитивно понятный алгоритм, который позволяет найти повторяющийся элемент всего за один проход по массиву. Его временная сложность составляет O(n).