|  |  |  |
| --- | --- | --- |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное автономное  образовательное учреждение высшего образования  «Пермский государственный национальный  исследовательский университет» | | |
|  | Институт компьютерных наук и технологий | |
| **ОТЧЁТ**  по индивидуальной работе №2  по дисциплине «Язык программирования C++»  Вариант 11 | | |
|  | | Работу выполнил  студент группы ИТ-11,12-2024 1 курса  Демин М.В  «24» Июня 2025 г. |
| Работу проверил  Кнутова Н.С  «26» Июня 2025 г. |
| Пермь 2025 | | |

**Содержание:**   
1.Постановка задачи ....................................................................................... 3

2.Алгоритм решения ....................................................................................... 4

3.Тестирования ................................................................................................ 7

4.Код программы .............................................................................................12

# Постановка задачи

Баржа. На барже располагается K грузовых отсеков. В каждый отсек можно поместить некоторое количество бочек с одним из 10 000 видов топлива. Причём извлечь бочку из отсека можно лишь в случае, если все бочки, помещённые в этот отсек после неё, уже были извлечены. Таким образом, в каждый момент времени в каждом непустом отсеке имеется ровно одна бочка, которую можно извлечь, не трогая остальных. Будем называть такие бочки крайними.

Изначально баржа пуста. Затем она последовательно проплывает через N доков, причём в каждом доке на баржу либо погружается бочка с некоторым видом топлива в некоторый отсек, либо выгружается крайняя бочка из некоторого отсека. Однако, если указанный отсек пуст, либо если выгруженная бочка содержит не тот вид топлива, который ожидалось, следует зафиксировать ошибку. Если на баржу оказывается погружено более P бочек или если после прохождения всех доков она не стала пуста, следует также зафиксировать ошибку. От вас требуется либо указать максимальное количество бочек, которые одновременно пребывали на барже либо зафиксировать ошибку.

Входные данные: в первой строке три целых числа N, K и P (1 ≤ N, K, P ≤ 100 000). Далее следует N строк с описанием действия, выполняемого в очередном доке. Если в нём происходит погрузка, то строка имеет вид «+ A B», где A — номер отсека, в который помещается бочка, а B — номер вида топлива в ней. Если же док занимается разгрузкой, то строка имеет вид «- A B», где A — номер отсека, из которого извлекается бочка, а B — номер ожидаемого вида топлива. Выходные данные: вывести либо одно число, равное искомому максимуму в случае безошибочного прохождения баржой маршрута, либо вывести слово «Error» в противном случае.

# Алгоритм решения

Введение:

Основные идеи и принципы работы:

Модель отсеков как стеков  
Каждый из K отсеков реализован как отдельный стек. Это помогает

лучше понять принцип работы баржи:

При погрузке бочка помещается на вершину стека соответствующего отсека

При разгрузке извлекается только верхняя бочка из указанного отсека

Динамическое отслеживание состояния  
Программа в реальном времени отслеживает:

currentBarrels: текущее количество бочек на барже

maxBarrelsEver: максимальное количество бочек за всё время работы

errorOccurred: флаг ошибки (прекращает обработку при установке)

Механизм валидации операций  
Перед выполнением каждой операции проверяется:

Корректность номера отсека (1 ≤ A ≤ K)

Соответствие типа топлива при разгрузке

Пустой ли отсек при разгрузке

Непревышение лимита P при погрузке

Выбор структур данных

Класс StackNode:  
 Базовый элемент стека, содержащий:

fuelType (тип топлива)

Указатель next на следующий элемент

Класс Stack:  
 Реализация стека с операциями:

push(): добавление элемента на вершину (O(1))

pop(): удаление элемента с вершины (O(1))

top(): получение значения верхнего элемента (O(1))

isEmpty(): проверка пустоты стека (O(1))

Класс Barge:  
 Основа системы, содержащее:

Массив стеков compartments для отсеков

Счётчики currentBarrels и maxBarrelsEver

Методы load() и unload() с валидацией

Управление памятью в деструкторе

# Алгоритм решения

Шаг 1: Подготовка и инициализация

Ввод и проверка параметров:

Программа запрашивает три числа:

N - общее количество операций (1-100,000)

K - количество отсеков на барже (1-100,000)

P - максимальное число бочек (1-100,000)

Проверяется, что все значения находятся в допустимом диапазоне

При неверном вводе запрашиваются значения повторно

Создание модели баржи:

Создается массив из K стеков (по одному на каждый отсек)

Инициализируются счетчики:

current - текущее количество бочек (изначально 0)

max - максимальное количество бочек за все время (изначально 0)

error - флаг ошибки (изначально false)

Шаг 2: Обработка операций

Для каждой из N операций:

Чтение данных операции:

Считывается символ операции (+ или -)

Считывается номер отсека (A)

Считывается тип топлива (B)

Выполнение операции:

Если операция погрузки (+):

Проверяется, что номер отсека корректен (1 ≤ A ≤ K)

Проверяется, что не превышен лимит бочек (current < P)

Если проверки пройдены:

Бочка добавляется в соответствующий отсек (стек)

Увеличивается счетчик current

Обновляется max (если current > max)

Если операция разгрузки (-):

Проверяется, что номер отсека корректен (1 ≤ A ≤ K)

Проверяется, что отсек не пуст

Проверяется, что тип топлива на вершине стека совпадает с B

Если проверки пройдены:

Бочка удаляется из отсека

Уменьшается счетчик current

Обработка ошибок:

При любой ошибке устанавливается флаг error = true

После ошибки остальные операции пропускаются

Шаг 3: Завершение работы

После обработки всех операций:

Проверяется:

Был ли флаг ошибки установлен (error == true)

Остались ли бочки на барже (current != 0)

Если хотя бы одно условие выполняется - выводится "Error"

Иначе выводится максимальное количество бочек (max)

Тестирование

1. Без ошибок

Входные данные:  
6 1 2

+ 1 1

+ 1 2

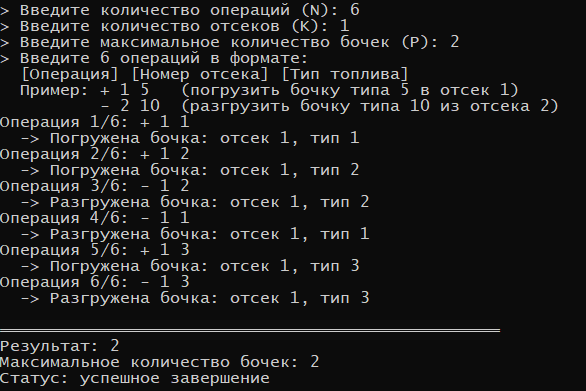
- 1 2

- 1 1

+ 1 3

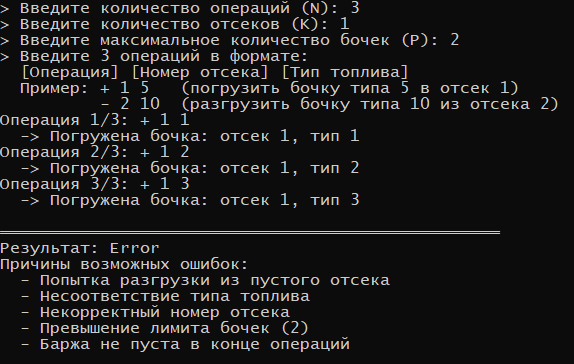
- 1 3

Ожидаемый результат : 2

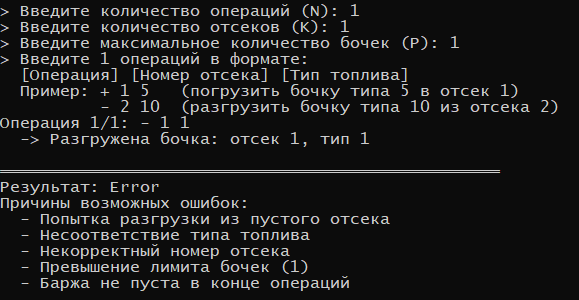


Всё верно, максимальное число бочек равно 2.

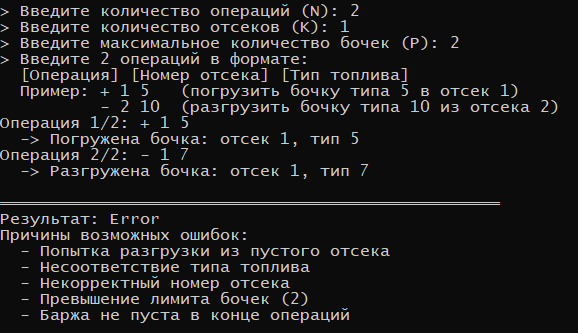
2.Ошибка (Количество бочек превышает лимит)



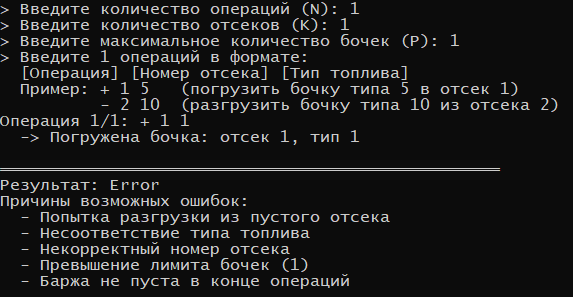
3.Ошибка (Убирают из пустого отсека)



4.Ошибка (Другой тип бочки)



5.Ошибка (Осталась 1 бочка)



6. Без ошибок с другими данными

Входные данные:

4 2 3

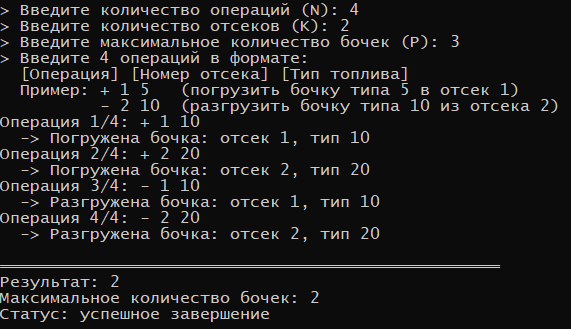
+ 1 10

+ 2 20

- 1 10

- 2 20

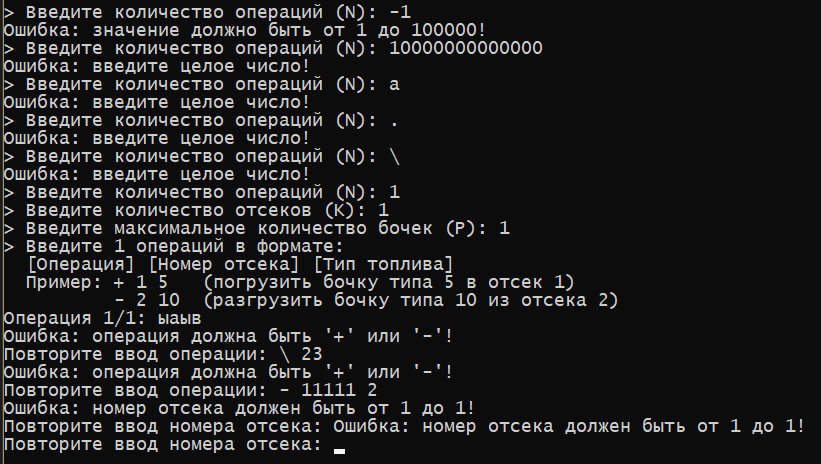
Ожидаемый результат: 2



Всё верно.

Максимальное число бочек равно 2, как и в ожидаемом результате

7.Ошибки при вводе данных (буквы, символы, числа). Защита от дурака.



Проверялись как числа, так и операнды для ввода.

Также проверки на не подходящие для программы числа, символы и буквы

Защита от дурака работает корректно

# Код программы:

<https://github.com/qwe9877978/Barge_IKM_11>