

# Название темы работы (Из Задания)

Исполнитель: Фамилия Имя Отчество, гр. ПИ-1-19

Представление работы должно занимать не более 3-4 минут. Поэтому выносим в презентацию только ключевые моменты исследования. Что для вас является ключевым – решайте сами. Приведённый здесь пример – лишь один из вариантов. Помним, что в среднем на каждый слайд уходит полминуты. Перед защитой отрепетируйте выступление. Останавливать буду жёстко при нарушении регламента.

# **Постановка задачи [закljučается в следующем]**

Разработать программный проект для реализации операций над обыкновенными дробями.

Программа должна работать с дробью, как с объектом, описываемым двумя числами – числитель и знаменатель.

Запрограммировать операции сложения, вычитания, умножения и деления дробей. Результат должен возвращать обыкновенную дробь.

# Математическое описание задачи

Рассмотрим алгебраические операции над двумя дробями

Пусть  $x_1 = \frac{n_1}{d_1}$  и  $x_2 = \frac{n_2}{d_2}$ ;

Сложение:  $x_3 = \frac{n_1}{d_1} + \frac{n_2}{d_2} = \frac{d_1 n_2 + d_2 n_1}{d_1 d_2} = \frac{n_3}{d_3}$ ; (1)

Умножение:  $x_4 = \frac{n_1}{d_1} \times \frac{n_2}{d_2} = \frac{n_1 \times n_2}{d_1 \times d_2} = \frac{n_4}{d_4}$ ; (2)

Деление:  $x_5 = \frac{n_1}{d_1} : \frac{n_2}{d_2} = \frac{d_2 \times n_1}{d_1 \times n_2} = \frac{n_5}{d_5}$ ; (3)

Анализ операций над обыкновенными дробями показывает, что все они оставляют результат в том же множестве обыкновенных дробей.

На основании ур-ий (1-3) нами были составлены числовые контрольные примеры, которые мы использовали для тестирования разрабатываемого проекта. В контрольных примерах рассмотрены все возможные исходы

# Приёмы алгоритмизации задачи

1. **Математические свойства** дробей позволяют нам создать структурную переменную `fraction`, которая состоит из двух полей целого типа `n` (числитель – `numerator`) и `d` (знаменатель – `denominator`).
2. Вводим **ограничения на разрабатываемую модель**:
  - а) мы не приводим сокращение дроби - эта сложная операция может быть разработана позднее.
  - б) целые числа рассматриваются как дробь с единичным знаменателем
  - в) правильные и неправильные дроби не классифицируются в проекте.
3. Для удобного **пользовательского интерфейса** мы произведём перегрузку операторов, коды которых будут реализованы в соответствии с уравнениями (1-3).

# Список разрабатываемых функций

`struct fraction(){int a,b};` - структура обыкновенная дробь

`fraction Input_f(string text = "")` – функция ввода дроби с консоли,  
`text` – комментарий – что вводить (по умолчанию пробел)

`fraction Output_f(string text="fraction = ")` – функция вывода дроби на консоль с поясняющим текстом - `text`.

`fraction operator+(fraction t);` - перегруженный оператор сложения, реализующий формулу (1).

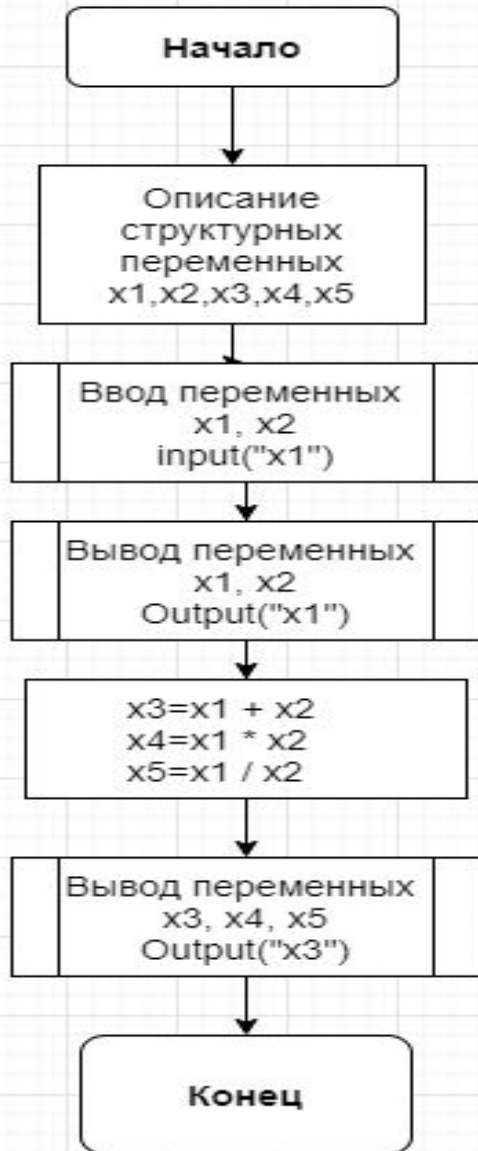
...

`fraction operator=(fraction t);` - перегруженный оператор присваивания.

....

Не более двух слайдов

# Блок-схема главной программы



## Ход соответствующей программы

```
int main()
{
fraction x1, x2, x3, x4, x5;
x1 = input("\n x1 =");
x2= input("\n x2 =");
x3 = x1 + x2;
x4 = x1 * x2;
x5 = x1 / x2;
output("x3=x1+x2=");
output("x4=x1*x2=");
output("x5=x1/x2=");
return 0;
}
```

В выводах надо постараться отразить основные достоинства, алгоритмические находки вашего исследования. Если проект сложный, то можно этому посвятить два-три слайда.

Выводы:

1. Код главной программы имеет линейную структуру, значит, разработанный проект удовлетворяет требованиям структурного подхода.
2. Внедрение структурной переменной обеспечило запись программы верхнего уровня приближенной к обыкновенной алгебраической записи.

# Результаты тестирования программы (для коротких результатов)

Числовой контрольный пример      Скриншот решения

Вывод: Результаты расчётов по программе полностью воспроизводят контрольный пример, следовательно программа работает верно и может быть применена для массовых расчётов.

# **Результаты тестирования программы (для результатов с большим объёмом информации)**

Можно разбить скрин на два- три слайда, не останавливаясь подробно на деталях, рассказать, что представлено на скрине.



# Заключение

1. В работе представлены математические основы для операций с обыкновенными дробями.
2. Войство дробей и операций над ними позволило нам создать структурную переменную `fraction`
3. На её основе реализовать перегрузку операторов для программируемых операций.
4. В результате код программ верхнего уровня приближен к алгебраической записи выражений
5. Программа прошла успешное тестирование при сравнении с контрольным примером. Она отлажена и может быть применена в массовых расчётах.

# Спасибо за внимание

ПОСТАРАЙТЕСЬ УЛОЖИТЬСЯ В 3 минуты. Крайний случай 4 минуты.