АТД **Дробь** SimpFrac.

Поля:

* Числитель numerator
* Знаменатель denominator

Конструктор:

Начальные значения:

1. numerator = 0 (float); // числитель

2. denominator = 1 (float); // знаменатель

Операции:

* Ввести числитель

Вход: new\_numerator

Предусловие: -

Процесс: numerator = new\_numerator

Выход: -

Постусловие: -

* Ввести знаменатель

Вход: new\_ denominator

Предусловие: - new\_denominator > 0

Процесс: denominator = new\_denominator

Выход: -

Постусловие: -

* Сложение

Вход: term

Предусловие: -

Процесс: numerator = numerator \* term.denominator + + denominator \* term.numerator

denominator \*= term.denominator

Сокращение дроби

Выход: -

Постусловие: -

* Вычитание

Вход: subtrahend

Предусловие: -

Процесс: numerator = numerator \* term.denominator - - denominator \* term.numerator

denominator \*= term.denominator Выход: -

Сокращение дроби

Постусловие: -

* Умножение

Вход: factor

Предусловие: -

Процесс: numerator \*= factor

Сокращение дроби

Выход: -

Постусловие: -

* Деление

Вход: divider

Предусловие: divider <> 0

Процесс: denominator \*= divider

Сокращение дроби

Выход: -

Постусловие: -

* Сравнение

Вход: comp

Предусловие: -

Процесс: if numerator / denominator > comp then answer = ‘Изначальная дробь больше’

if numerator/ denominator < comp

then answer = ‘Изначальная дробь меньше’

if numerator/ denominator – comp <= 0.0000001

then answer = ‘Числа равны’

Выход: answer

Постусловие: -

* Перевод в десятичную дробь

Вход: -

Предусловие: -

Процесс: -

Выход: numerator / denominator

Постусловие: -

* Сокращение дроби

Вход: -

Предусловие: -

Процесс: Нахождение НОД по алгоритму Евклида

numerator /= НОД

denominator /= НОД

Выход: -

Постусловие: -