

Лабораторная работа
Создание финансовых калькуляторов.
Использование чисел с фиксированной запятой

В данной работе следует выполнить и прислать на почту (сдать в компьютерном классе):

1. Запрограммировать финансовый калькулятор для четырех операндов:

Число1	$\begin{matrix} + \\ - \\ * \\ / \end{matrix}$	(Число2	$\begin{matrix} + \\ - \\ * \\ / \end{matrix}$	Число3	$\begin{matrix} + \\ - \\ * \\ / \end{matrix}$	Число4)	$\begin{matrix} + \\ - \\ * \\ / \end{matrix}$	Число5	=	Вид округления? математическое бухгалтерское усечение	Итоговый результат
--------	--	---	--------	--	--------	--	--------	---	--	--------	---	---	-----------------------

Разрешены **четыре операции**: сложение, вычитание, умножение и деление.

Обеспечить **диапазон чисел**: не менее чем от -1 000 000 000 000.000 до +1 000 000 000 000.000. Числа 1–5 могут быть как положительными, так и отрицательными.

Если итоговый результат (или на промежуточном шаге вычислений) превышает указанный диапазон чисел, то допускается просто написать сообщение о переполнении. Так же обязательно писать о попытке деления на 0.

Количество значащих знаков после запятой при вводе от пользователя и выводе итогового результата всегда равно трем. При этом не значащие нули после запятой не отображаются!

Порядок расчетов. Все промежуточные вычисления из двух операндов округлять до 6 знаков после запятой **по правилам математического округления**. Полученный итоговый результат перед выводом пользователю необходимо округлять до 3 знаков после запятой. При этом калькулятор должен иметь опцию – **вид округления итогового результата**:

- а) математическое;
- б) бухгалтерское (банковское);
- в) усечение.

Приоритет операций – как принято в математике, при этом первые три числа (две операции) заключены в скобки, т.е. всегда имеют приоритет выполнения перед последней операцией.

Пример расчетов (приведен не конкретно для текущей задачи, а для пояснения округления промежуточных вычислений):

$(-7 / 9 * 987\,000\,789,987) + 767\,667\,500,006 =$ [делим -7 на 9, получаем -0,7777777..., округляем до 6-ти знаков и получаем -0,777778] =

$= -0,777778 * 987\,000\,789,987 + 767\,667\,500,006 =$ [в результате умножения получаем 767 667 500,4345089, округляем до 6-ти знаков и получаем 767 667 500,434509] =

$= -767\,667\,500,434509 + 767\,667\,500,006 = -0,428509 =$ [округляем итоговый результат до трех знаков после запятой и получаем]:

- | | |
|-----------------------------|----------|
| если выбрано математическое | = -0,429 |
| если выбрано бухгалтерское | = -0,428 |
| если выбрано усечение | = -0,428 |

Знаки перед запятой (целая часть) обязательно отображать всегда в разбивке пробелом по тысячам, миллионам, миллиардам.

Программа должна работать независимо от того, пользователь вводит в качестве разделителя точку или запятую. Допускается автоматически менять ее на любой единый «удобный» пользователю разделитель (точку или запятую).

2. Создать калькулятор платежей по кредиту. На вход подаются:

- a) дата получения кредита;
- b) фиксированная процентная ставка в месяц (*например, 1.25% в месяц*);
- c) начальная сумма кредита (*например, 1 000 \$*);
- d) фиксированный ежемесячный платеж (*например, 100.25 \$*).

Допустимый диапазон и все правила расчетов, как в финансовом калькуляторе выше, округление – математическое (отличие от калькулятора: перед выводом все числа округлять до 2-х знаков).

Требуется составить таблицу вида:

Дата начала периода	Дата конца периода	Сумма в начале	Сумма в конце периода с учетом процентов	Сумма после оплаты ежемесячного платежа
01.01.2015	30.01.2015	1 000.00	1 012.50	912.25
31.01.2015	01.03.2015	912.25	923.65	823.40
02.03.2015	31.03.2015	823.40	833.69	733.44
...

Таблица строится до момента, когда кредит будет полностью погашен, т.е. значение в «сумма после оплаты ежемесячного платежа» стало отрицательным (эту строку тоже отображаем); или за 60 временных периодов. Каждый период равен 30 **календарным** дням.

Даты должны быть представлены в виде ДД.ММ.ГГГГ. Все суммы в таблице выравнивать по правому краю, **всегда отображать два знака** после запятой уже после округления и отделять пробелами тысячи, миллионы, и т.д.

Колонка «Сумма после оплаты ежемесячного платежа» вычисляется как разность «Сумма в начале» и «Сумма в конце периода с учетом процентов». Колонка «Сумма в начале» равна значению «Сумма после оплаты ежемесячного платежа» предыдущего периода.

Обеспечить по клику мышкой (пальцем на смартфоне) сортировку столбцов таблицы. При этом даты должны сортироваться как даты, а числа как числа.

Порядок сдачи и оценки работы:

1. Сдается / демонстрируется работающее приложение. Исходные коды также обязательно демонстрируются и комментируются.
2. Работающим признается только приложение, в котором выполнены ВСЕ перечисленные требования. Если хоть одно из требований не выполнено – работа отправляется на доработку.
3. На экранных формах приложения должны отсутствовать ошибки (орфографические и синтаксические). Текст, таблицы, элементы управления на экранной форме должны быть выровнены между собой и выполнены в едином стиле. Переходы между элементами управления на форме должны быть упорядочены.
4. За полностью выполненную работу в срок **до 1 марта** включительно исполнитель получает 100 баллов. За каждую пророченную неделю – минус 10 баллов, но не менее 50 баллов (зачетный уровень по всем лабораторным в сумме 70 %). Не добрали баллы – будете делать дополнительные лабораторные работы.
5. Если выбран язык программирования (технология, среда, приложение), отличный от использованных другими студентами в уже зачетных работах, то дополнительно получите плюс 10% к баллам.
6. Изменение региональных настроек компьютера естественно не должны влиять на вычислительные результаты работы программы.