



И нформатика



Лекция №5. Тема: «Основы офисных пакетов.»



Написал диплом в **LaTeX**, а не в **Word**

Ожидание

- Никаких проблем с форматированием
- Картинки и таблицы сами нумеруются

Реальность

Ожидания сбылись. Но документы принимаются только в формате .docx



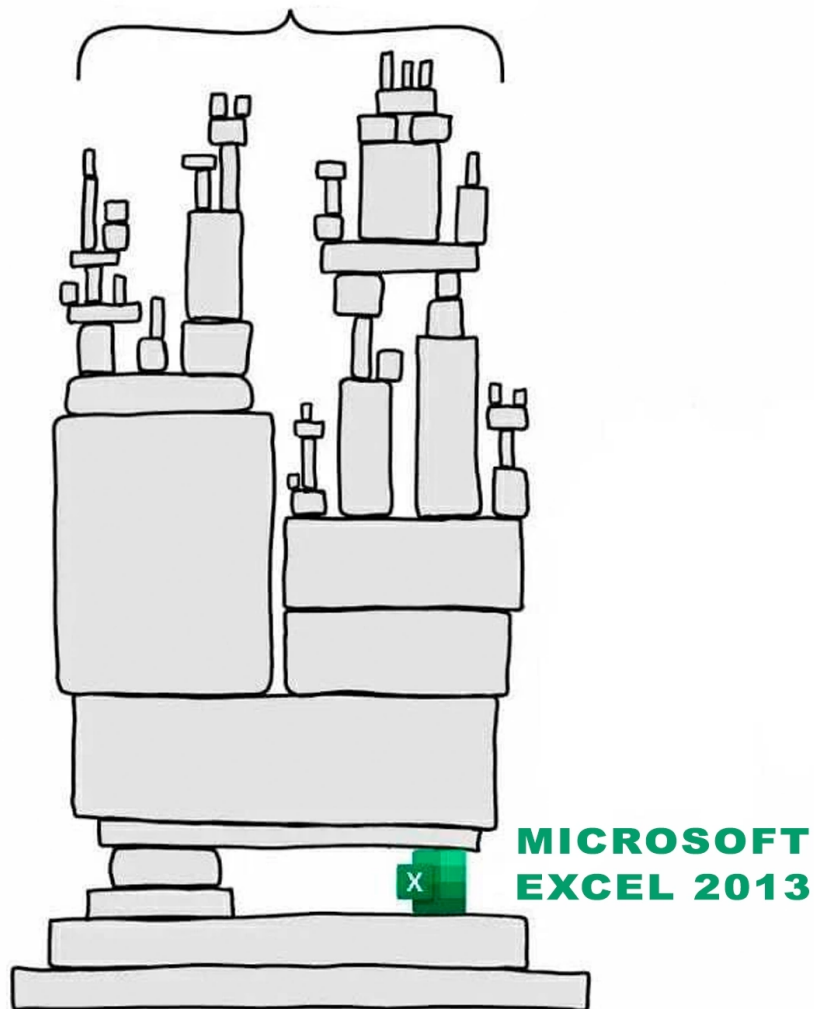
**Немного
передвинул
картинку
влево**

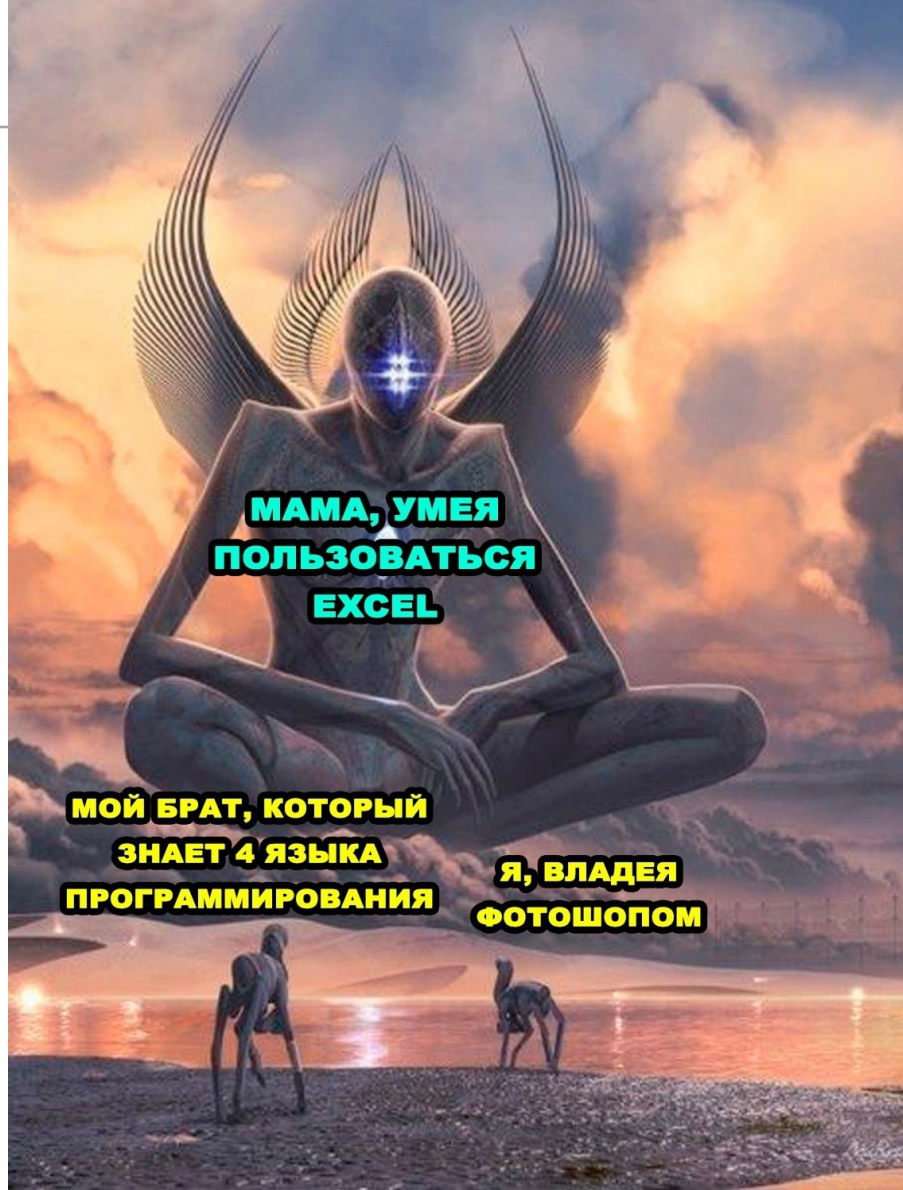


**Весь текст съехал, 4
новые страницы
открылись, орбита Земли
сместилась на 2 метра,
где-то вдалеке зазвучали
сирены**

ВСЯ МИРОВАЯ ФИНАНСОВАЯ СИСТЕМА

Немного шуток (2)





**МАМА, УМЕЯ
ПОЛЬЗОВАТЬСЯ
EXCEL**

**МОЙ БРАТ, КОТОРЫЙ
ЗНАЕТ 4 ЯЗЫКА
ПРОГРАММИРОВАНИЯ**

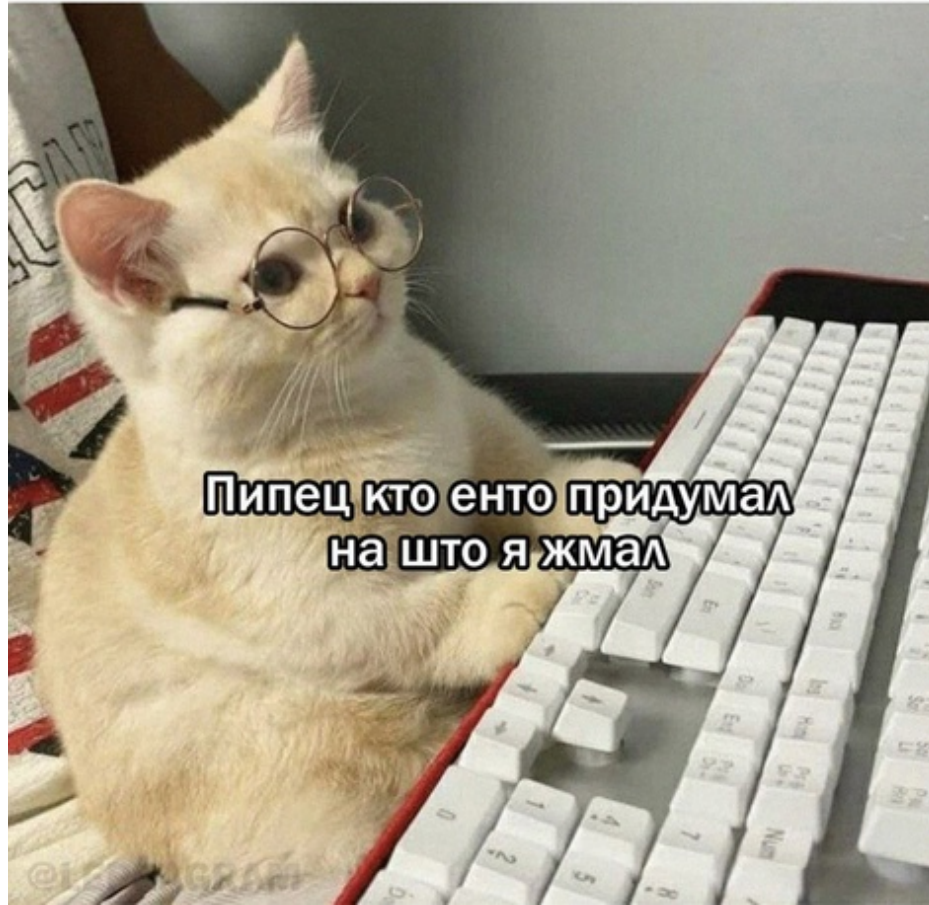
**Я, ВЛАДЕЯ
ФОТОШОПОМ**



Я в резюме: уверенный пользователь
Excel

Немного шуток (4)

Я на работе:



Пипец кто енто придумал
на што я жмал



К офисному программному обеспечению (ПО) относят наиболее часто применяемые в офисной работе программы для редактирования электронных документов. Существует более 30 серьёзных **офисных пакетов** разных производителей. Они различаются по составу и функциональности, но почти во всех присутствуют следующие три обязательных компонента:

- Текстовый процессор (текстовый редактор) – ТП.
- Электронная таблица (табличный процессор) – ЭТ.
- Программа подготовки презентаций – ПП.

Форматы файлов офисного ПО (наиболее популярные)

- ТП: doc, docx, odt
- ЭТ: xls,xlsx, ods
- ПП: ppt, pptx, odp

Интересные факты

- Форматы doc/xls/ppt до сих пор «закрыты» (по состоянию на 2017 год), хотя в разное время компания Microsoft предоставляла временный и/или частичный доступ к ним.
- Форматы docx, odt, xlsx, ods, pptx, odp – это zip-архивы с xml- и медиафайлами.
- Криптографическая защита в doc, xls, ppt крайне слабая (даже для длинных паролей).



Наиболее популярные офисные пакеты

Данные о популярности офисных пакетов получены с помощью анализа статистики, собранной с помощью сайта trends.google.com. В таблице пакеты приведены по убыванию популярности. Стоимость указана для desktop-версий.

Название офисного пакета	Особенности	Примерная стоимость на 2022 год, руб.	Исходный код
Google Docs, Яндекс.Диск, Облако Mail.ru	Ориентация на публичные облачные решения	бесплатно	закрытый
Microsoft Office	Имеет наиболее богатая функциональность , захватил > 90% desktop установок	5000–17000	закрытый
LibreOffice, OpenOffice, Calligra Suite	Слабая поддержка одновременного редактирования	бесплатно	открытый
iWork	Узкая ориентация на технику фирмы Apple	бесплатно	закрытый
WPS Office	Интерфейс идентичен Microsoft Office	3000-8000	закрытый
WordPerfect Office	Узкая ориентация на рынок персональных компьютеров	7000–28000	закрытый
OnlyOffice, Feng Office	Приоритетная ориентация на частные и публичные облачные решения	бесплатно*	открытый

Классификация офисных пакетов





Открытый бесплатный формат **ODF** (Open Document Format) позволяет обеспечить возможность долгосрочного хранения электронных документов без привязки к «капризам» конкретного производителя офисного ПО. Стандарты ODF описывают 16 форматов файлов (документы, картинки, таблицы, формулы, диаграммы), включая odt, ods, odp.

Стандартизация ODF в России (во многих других странах ситуация похожая)

- ODF 1.0 был описан и введён в действия по ГОСТ 26300-2010 (с 1 июня 2011 г.)
- ГОСТ 26300-2010 должен использоваться для документооборота в госструктурах.
- Стандартизация ODF не означает навязывание LibreOffice/OpenOffice.

Проблемы ГОСТ 26300-2010

- Текущая версия ODF уже 1.3 (в ней исправлены многие проблемы версии 1.0)
- Не описаны спецификации скриптов и макросов.
- Не описано применение цифровых подписей.
- Не описан язык описания формул.
- Не допускается использование таблиц в презентациях.

Спецификация ODF 1.3 (апрель 2021), принят 21 января 2020:

<http://docs.oasis-open.org/office/OpenDocument/v1.3/os/>

«Продвинутые» функции текстовых процессоров и электронных таблиц



В школе офисные пакеты изучаются очень подробно. Однако есть ряд немаловажных функций текстовых процессоров и электронных таблиц, о которых в школе почти не говорят.

Текстовый процессор

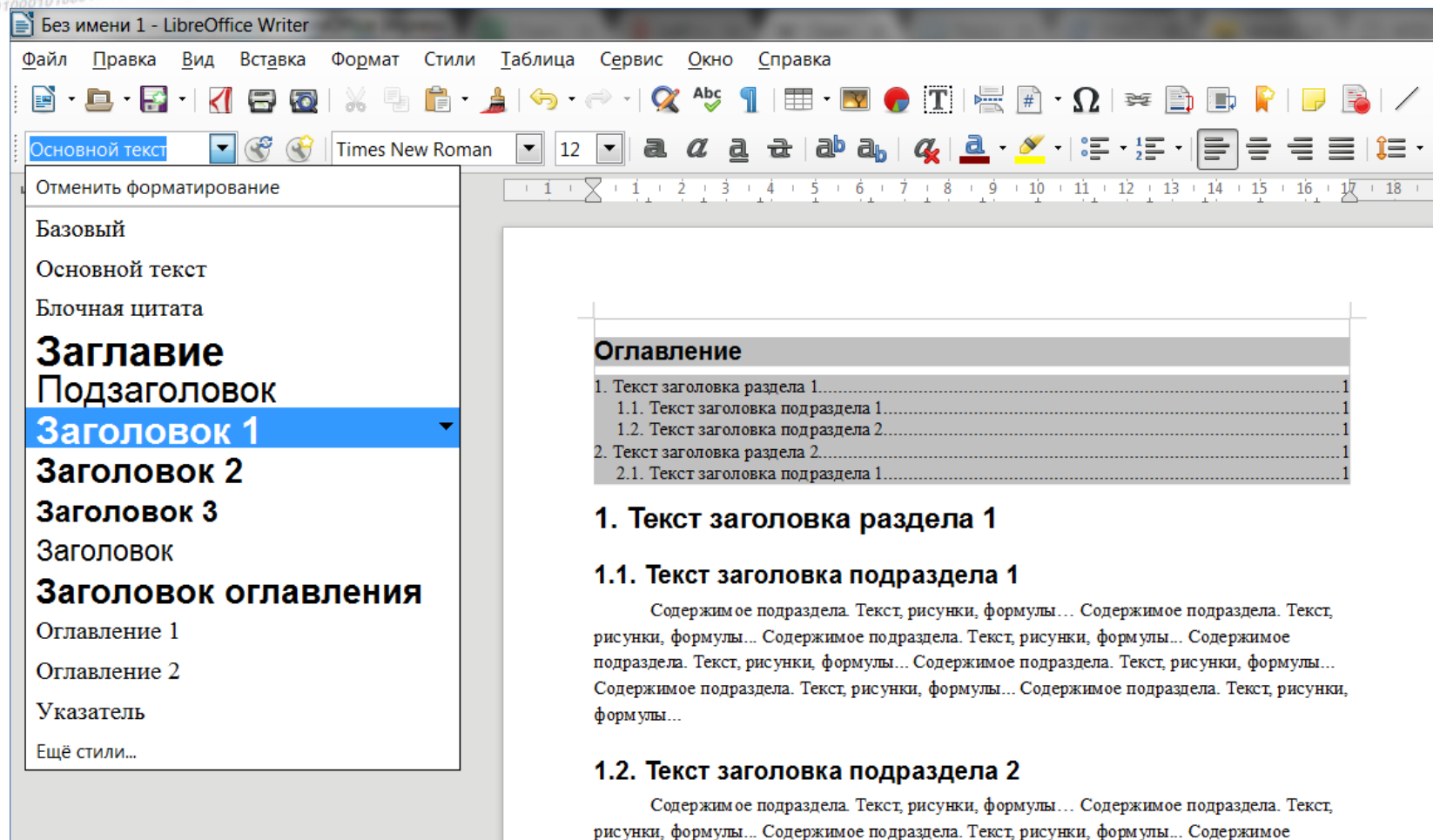
- Концепция стилей для оформления текстового документа
- Автонумерация рисунков, таблиц, формул
- Макросы для автоматизации повторяющихся действий
- Автозаполнение «мусорным» текстом

Табличный процессор

- Расчёт доверительного интервала
- Фильтры содержимого таблиц
- Запрет на ввод некорректных значений в ячейку.
- Условное форматирование
- Инструмент «Подбор параметра»

Рассматриваемые далее примеры выполнены в LibreOffice 5.1, однако в других офисных пакетах есть аналогичные функции (даже их названия почти всегда дословно совпадают).

Концепция стилей в текстовых процессорах



Автособираемое оглавление с помощью стилей

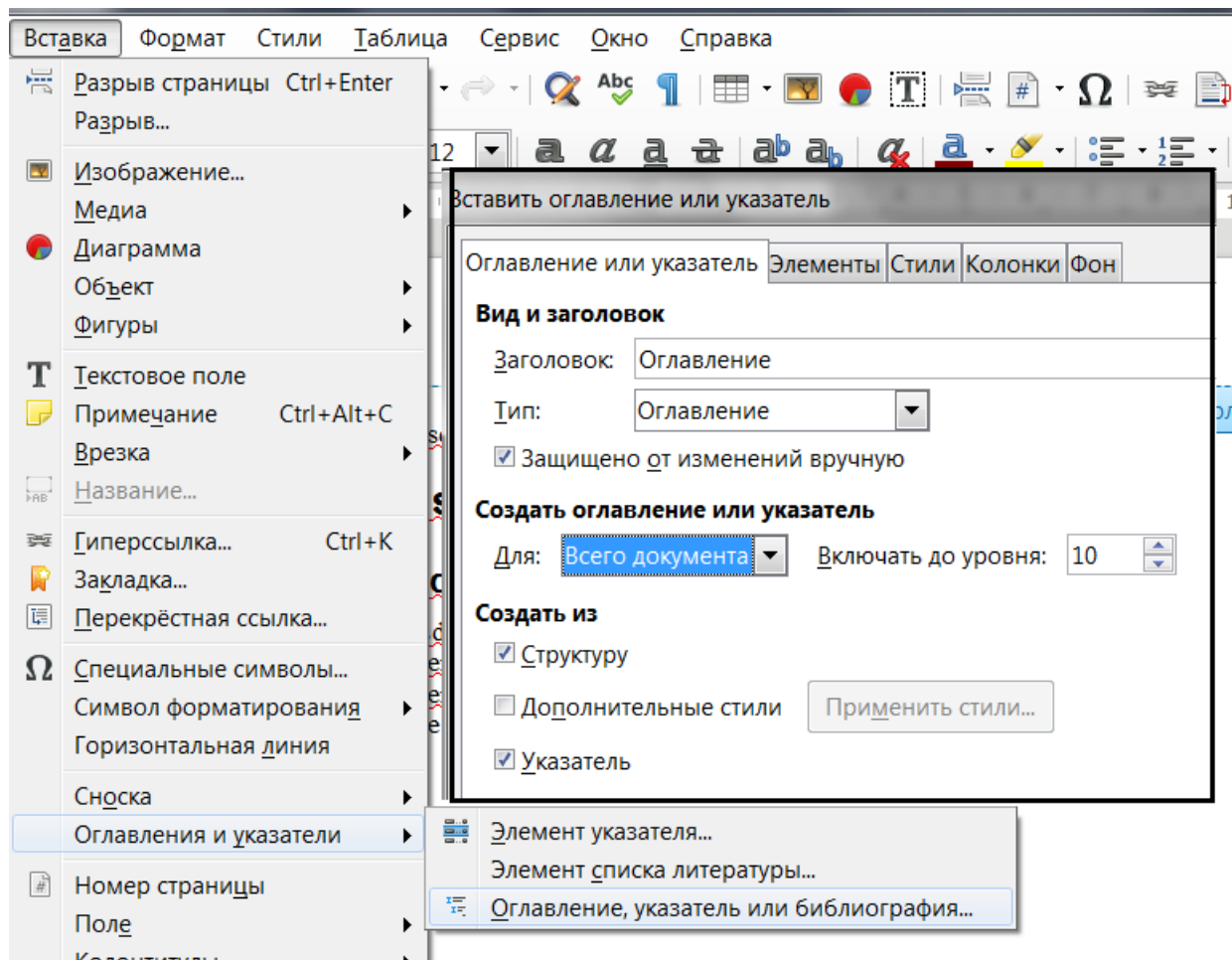


Алгоритм

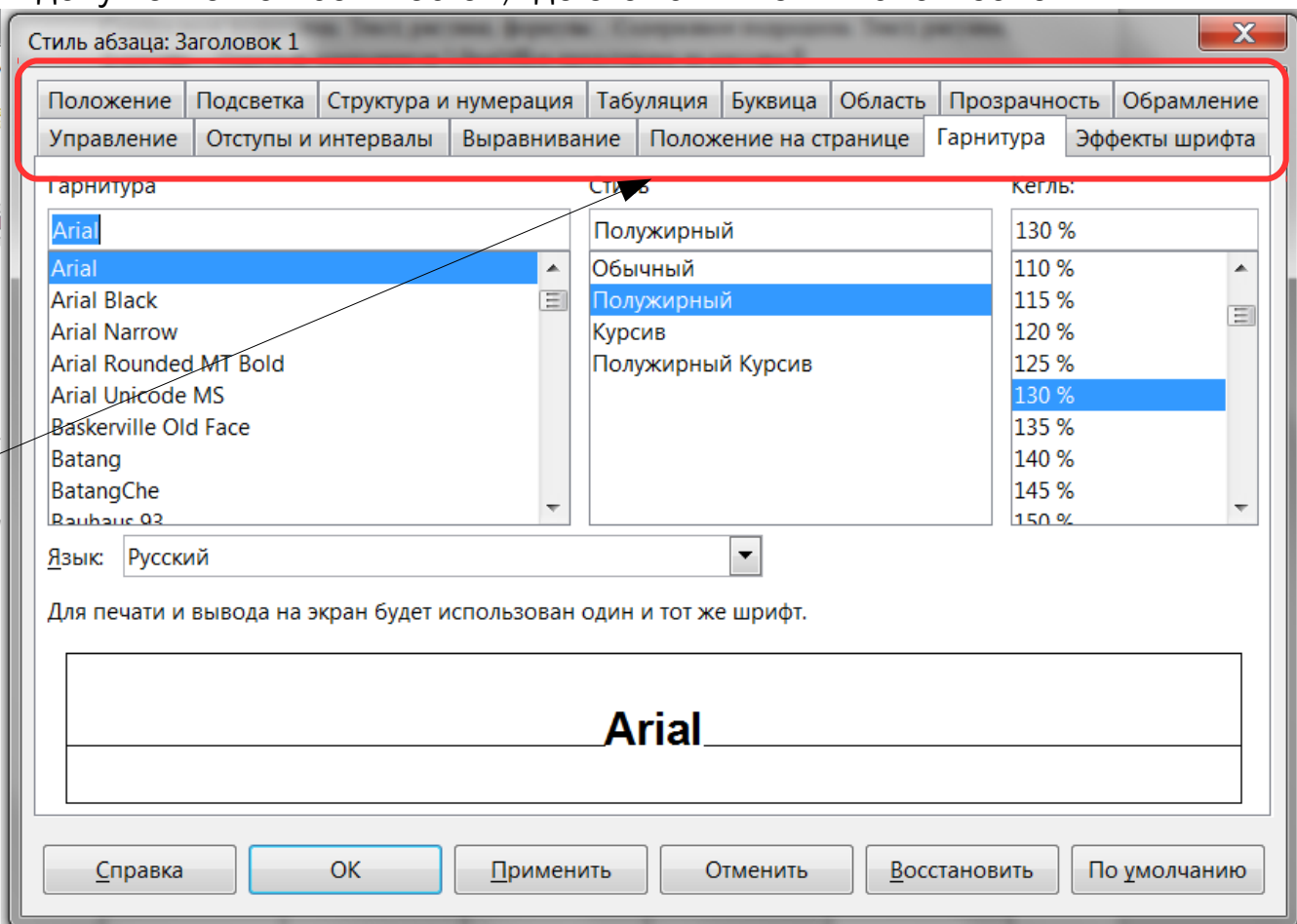
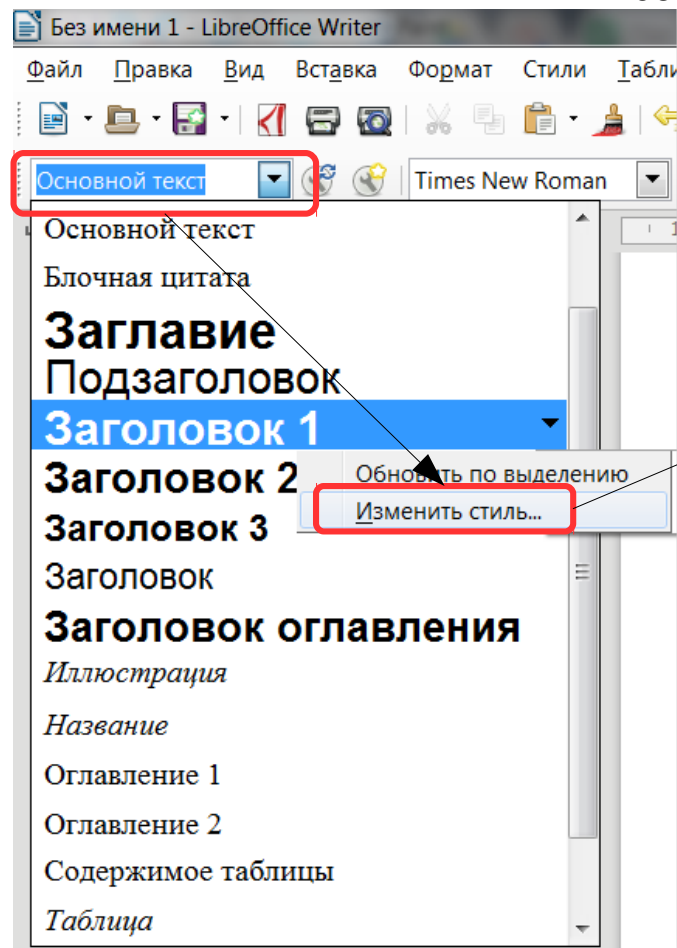
1. При первичном наполнении документа использовать **только** стили для разметки структуры текста.
2. Наполняя документ, не тратить время на оформление внешнего вида «буковок».
3. Приступить к настройке внешнего вида стилей только после окончательного наполнения документа текстом.

Не нужно форматировать текст вручную без стилей, задавая кегль, цвет шрифта и т. п. «врукопашную»!

Примечание. Приведённые рекомендации имеют смысл лишь при оформлении больших сложных документов!



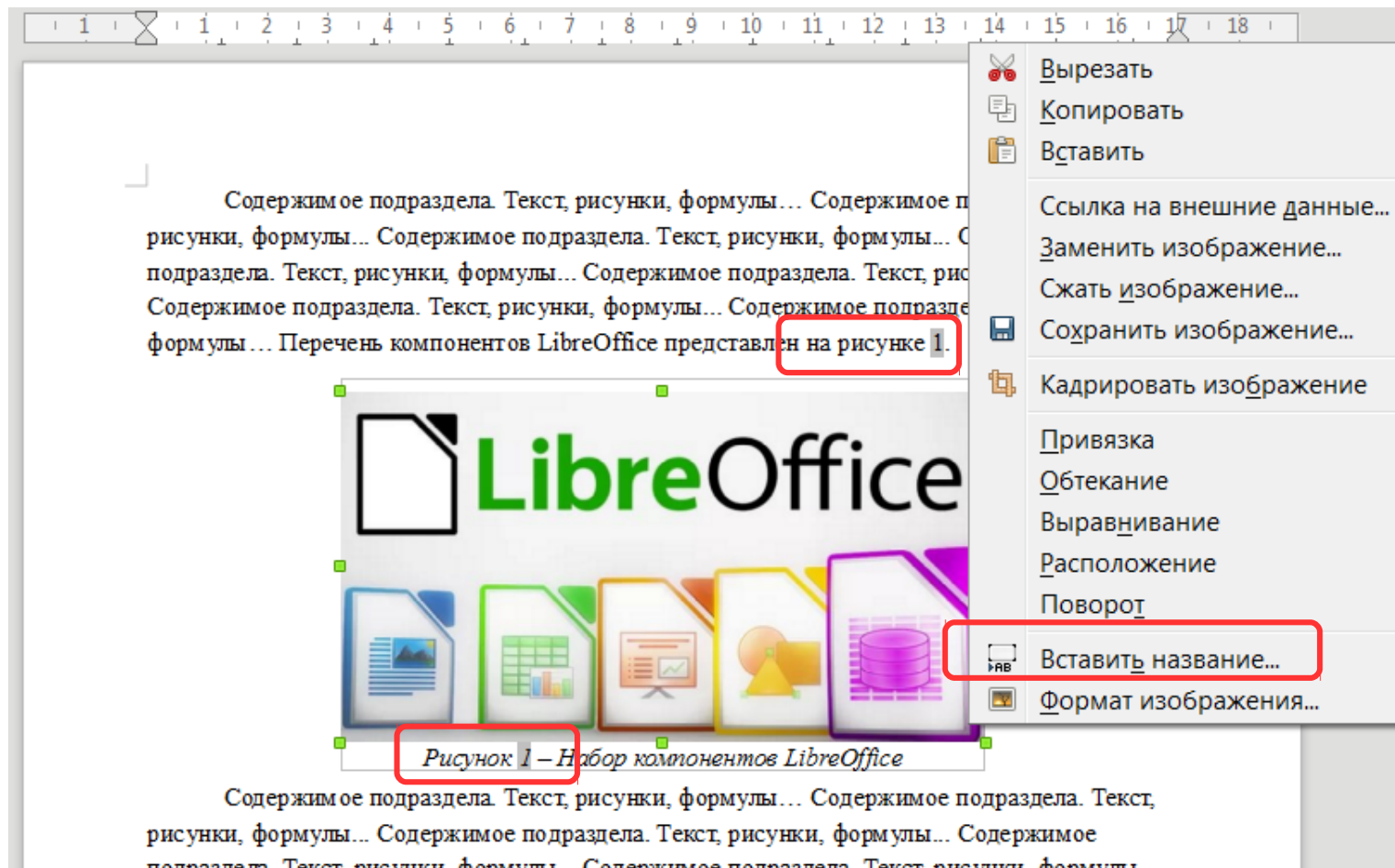
При изменении настроек стиля автоматически изменится отображение текста во всём документе во всех местах, где этот стиль был использован!



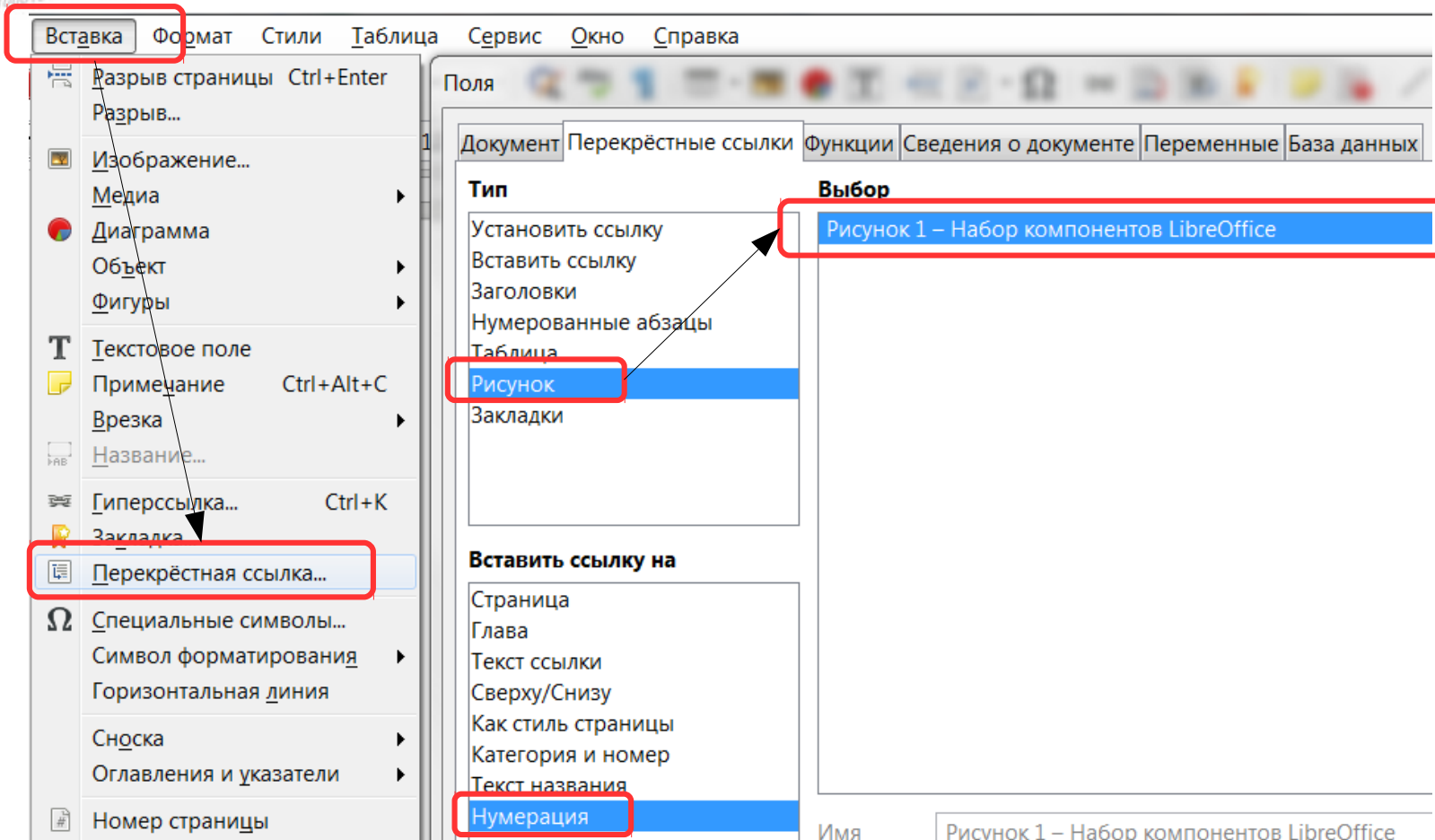
Перекрёстные ссылки и автонумерация рисунков

Памятка

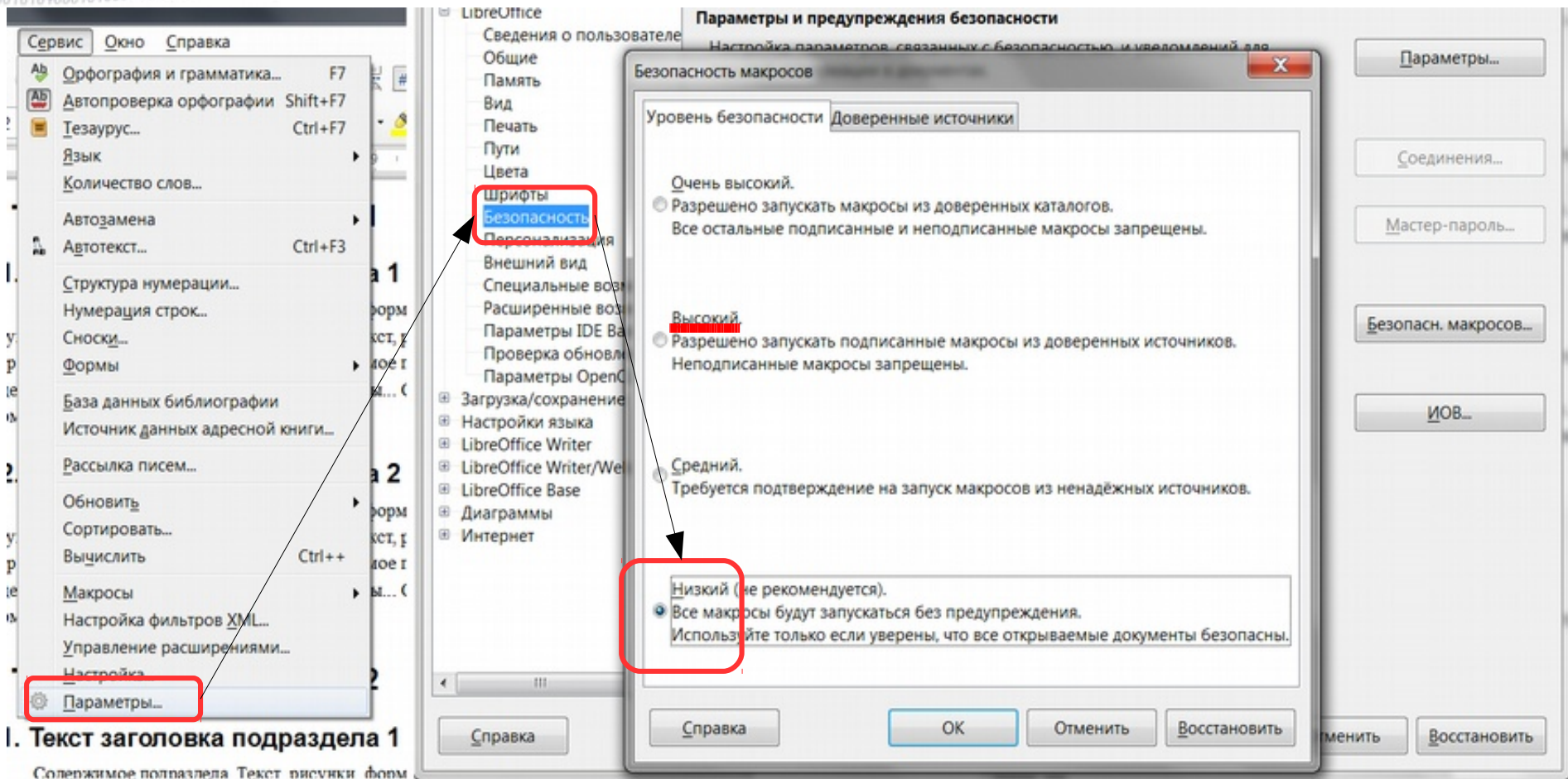
- При добавлении нового рисунка его порядковый номер будет выбран автоматически.
- При изменении порядка следования рисунков они автоматически перенумеруются
- Для принудительной перенумерации следует нажать F9 (или меню «Сервис → Обновить»).



Перекрёстные ссылки и автонумерация рисунков (2)



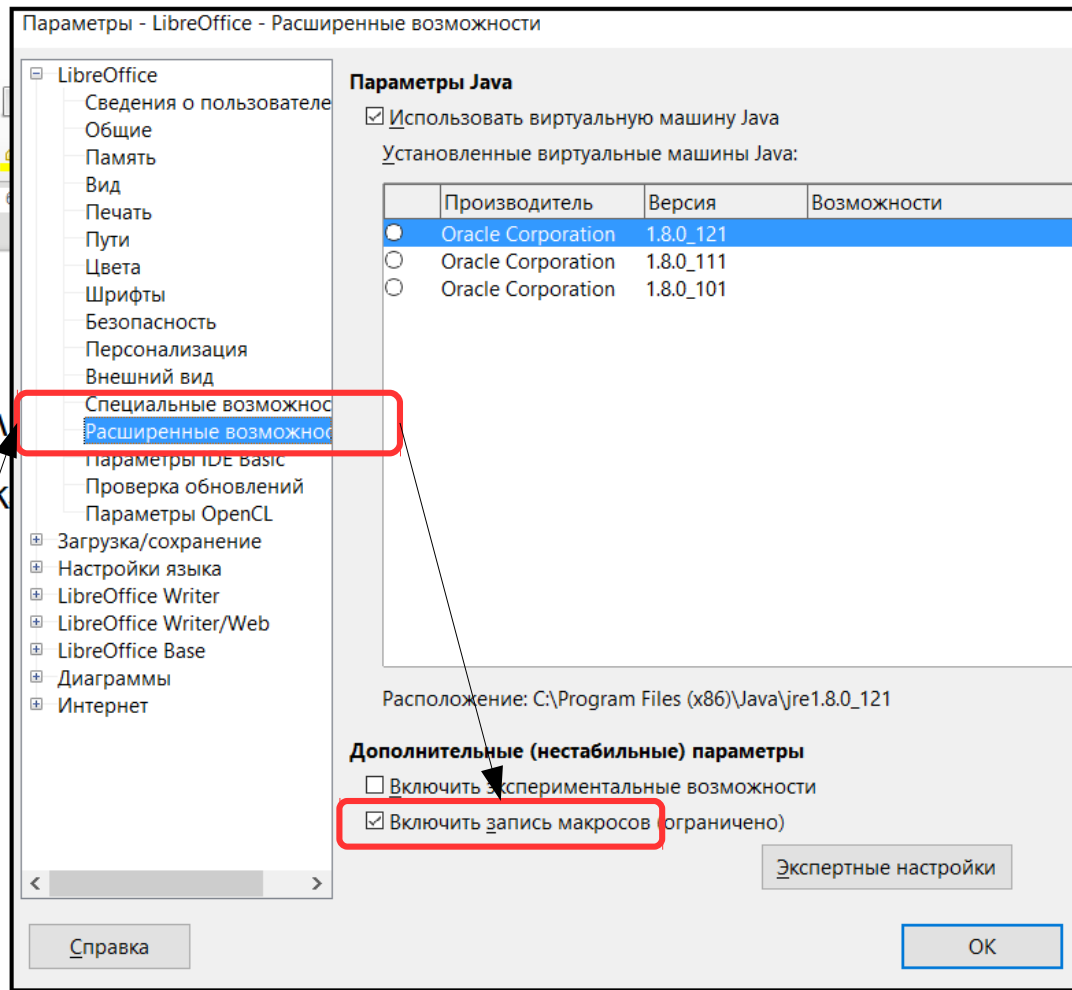
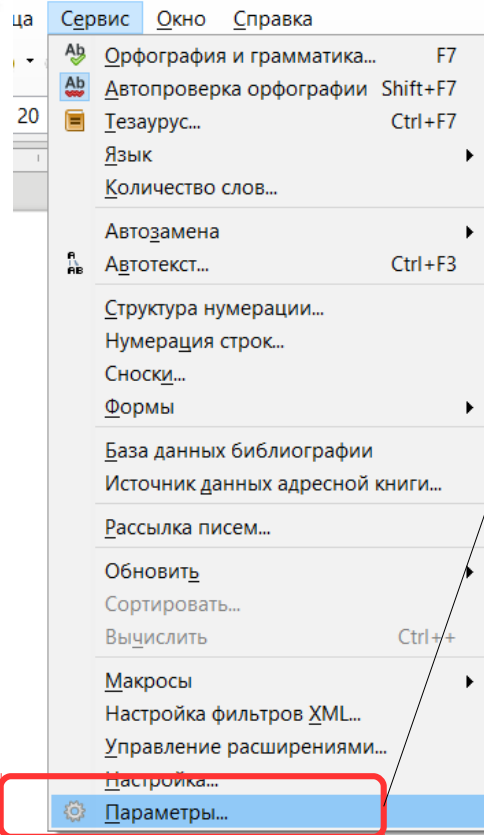
Макросы: особенности модели безопасности



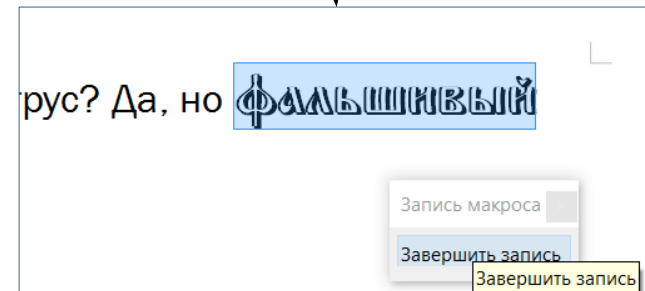
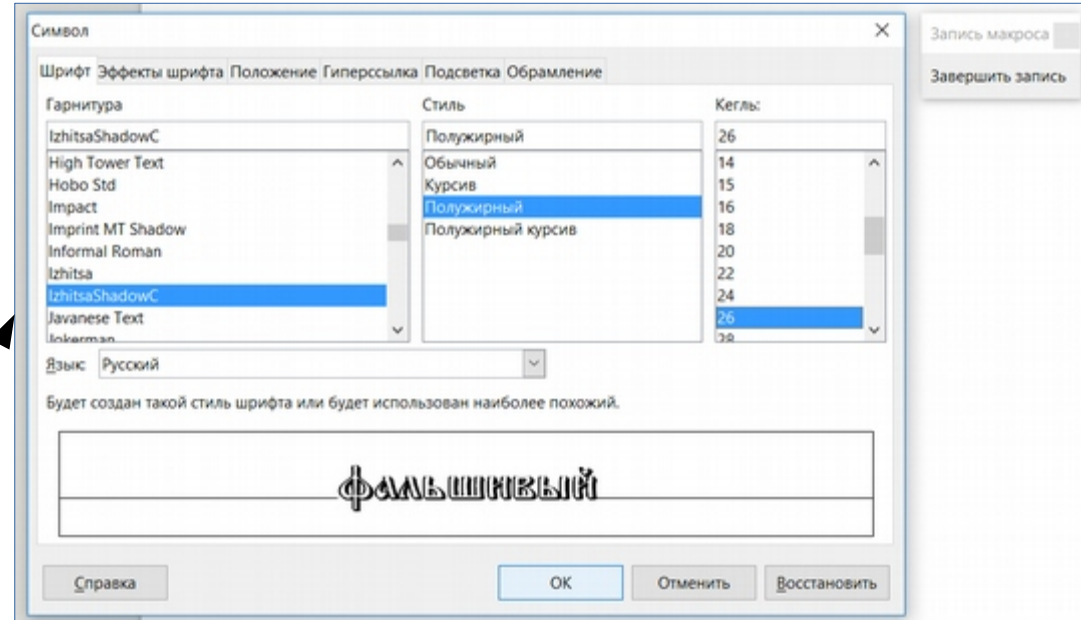
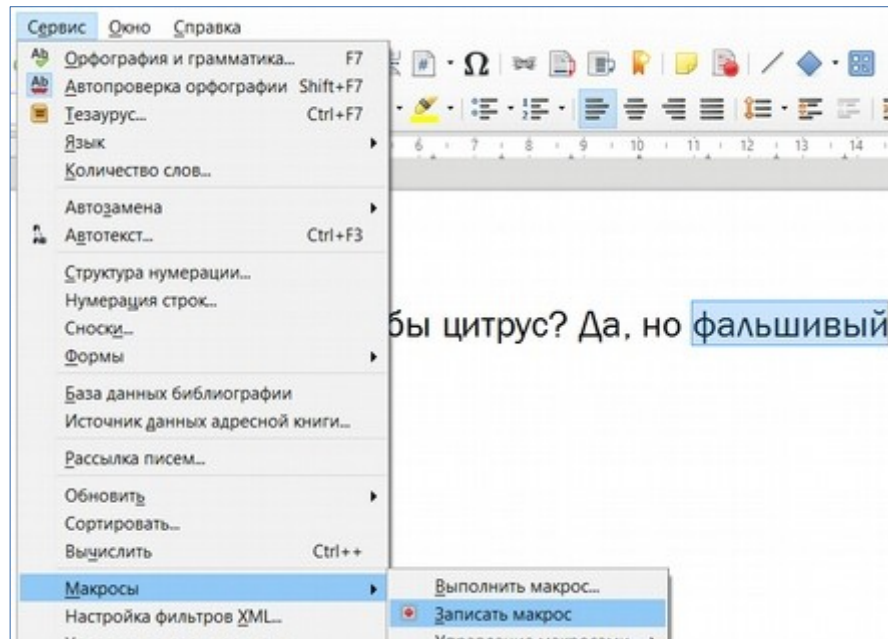
I. Текст заголовка подраздела 1

Содержимое подзаголовка Текст рисунка форм

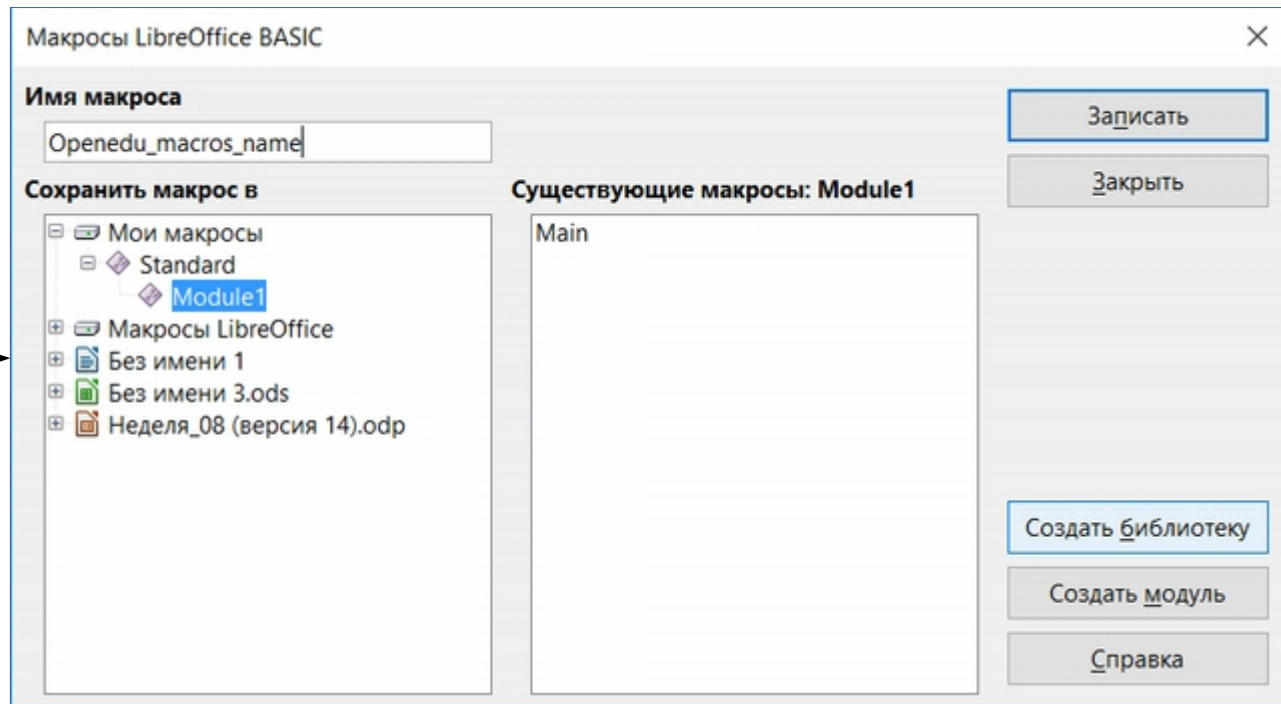
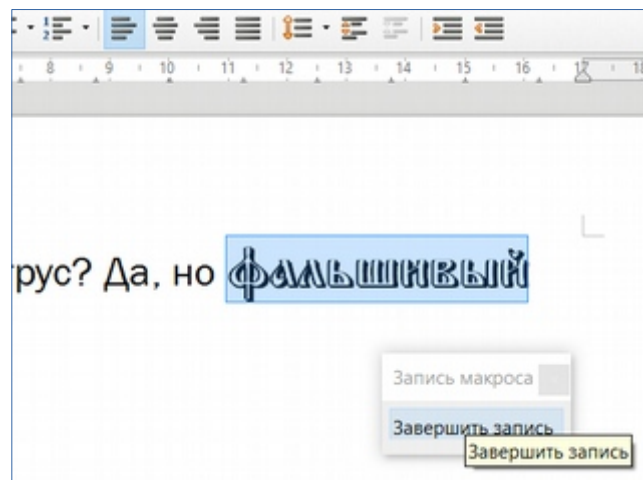
Макросы: запись вместо программирования вручную



Макросы: пример записи макроса



Макросы: пример записи макроса (продолжение 1)



Макросы: пример записи макроса (продолжение 2)

Сервис Окно Справка

- Орфография и грамматика... F7
- Автопроверка орфографии Shift+F7
- Тезаурус... Ctrl+F7
- Язык
- Количество слов...
- Автозамена
- Автотекст... Ctrl+F3
- Структура нумерации...
- Нумерация строк...
- Сноски...
- Формы
- База данных библиографии
- Источник данных адресной книги...
- Рассылка писем...
- Обновить
- Сортировать...
- Вычислить Ctrl++
- Макросы**
- Настройка фильтров XML...
- Управление расширениями...
- Настройка...
- Параметры...

Выполнить макрос...

Записать макрос

Управление макросами

Цифровые подписи...

Управление диалогами...

Выбор макроса

Выберите библиотеку, которая содержит нужный макрос. Затем выберите макрос в поле «Имя макроса».

Библиотека

- Мои макросы
- Standard
- Module1**
- WikiEditor
- Макросы LibreOffice
- Без имени 1

Имя макроса

- Main
- Openedu_macros_name**

Выполнить

Отменить

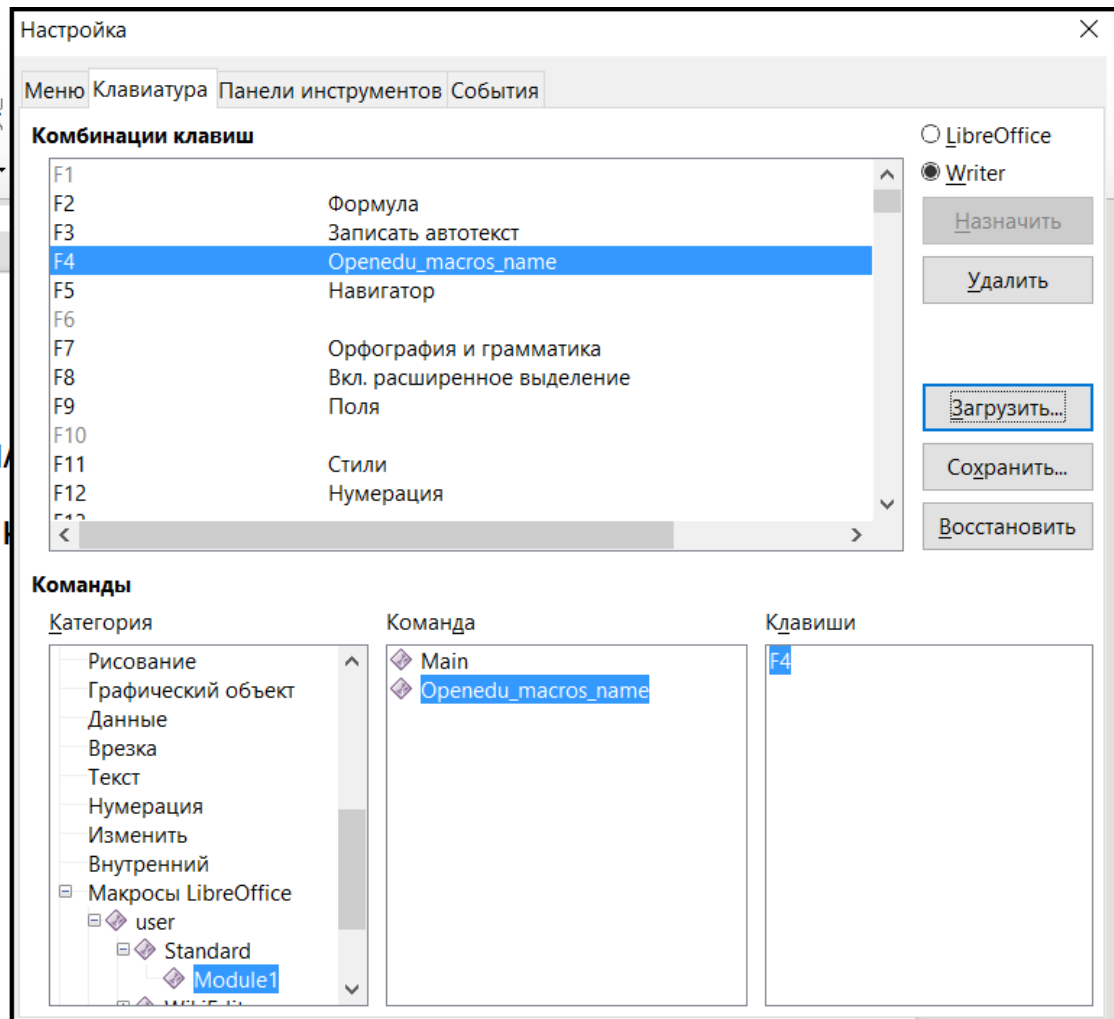
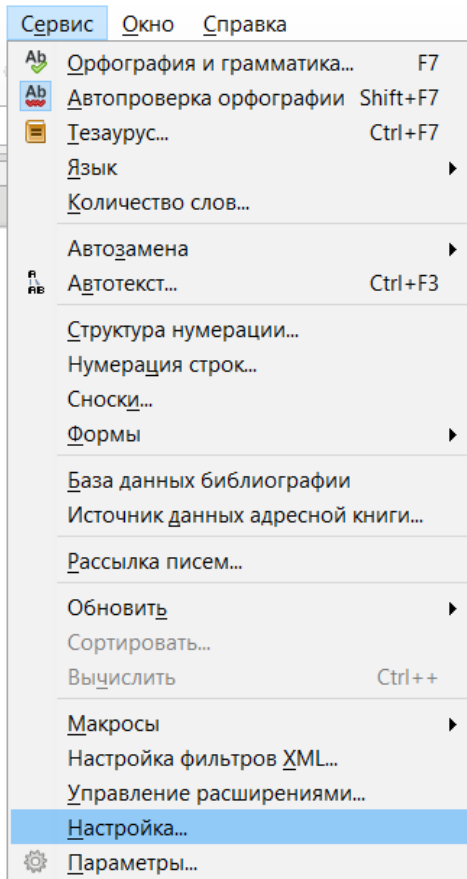
Справка

Описание

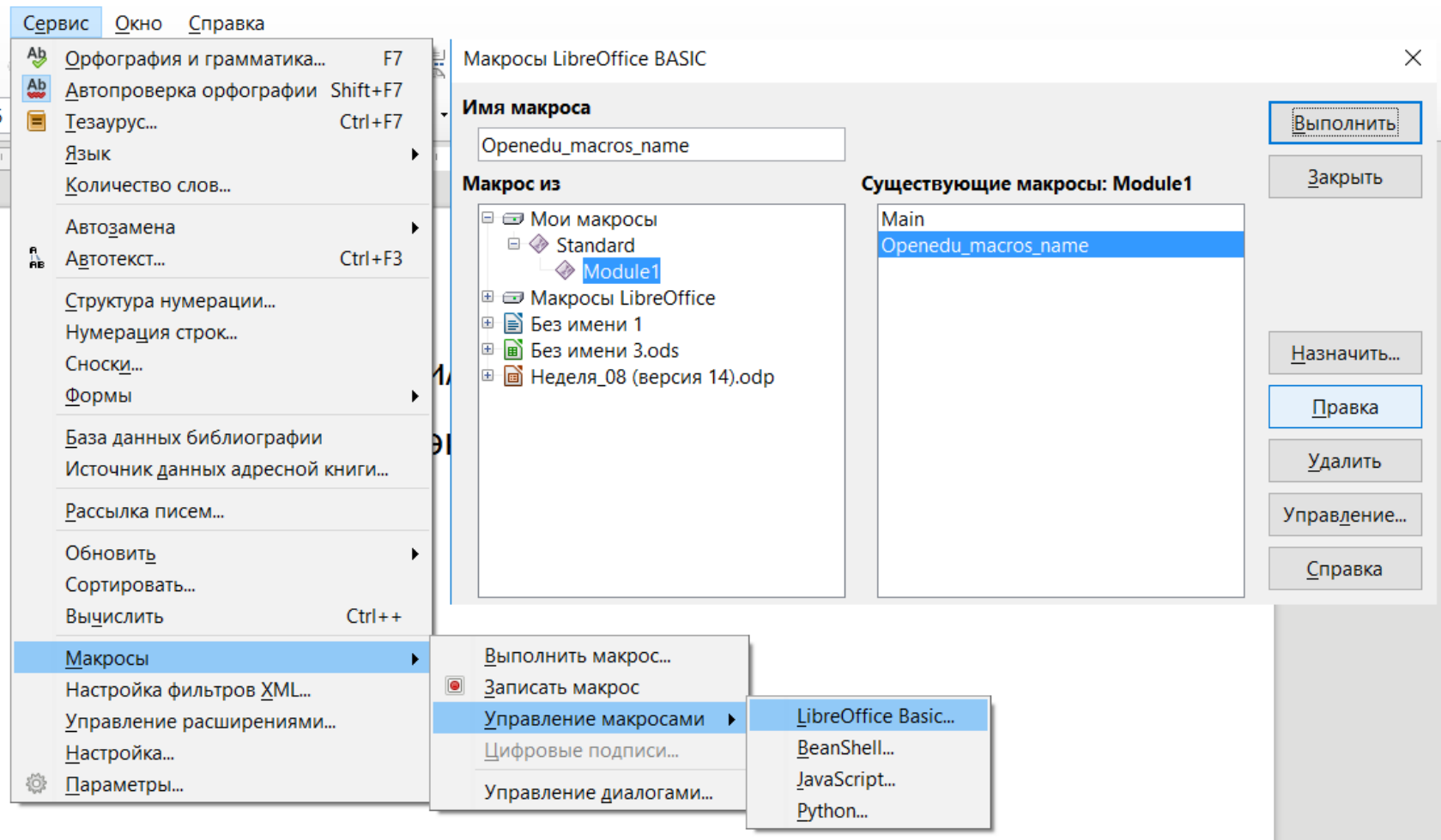
бы цитрус? Да, не экземпляр!

бы цитрус? Да, не экземпляр!

Макросы: назначение макроса на горячую клавишу



Макросы: как исправить макрос



The image shows the LibreOffice application interface with the **Сервис** (Tools) menu open. The **Макросы** (Macros) option is selected, which has opened a sub-menu. In this sub-menu, the **Управление макросами** (Manage Macros) option is highlighted, and its own sub-menu is visible, showing **LibreOffice Basic...** as the selected option.

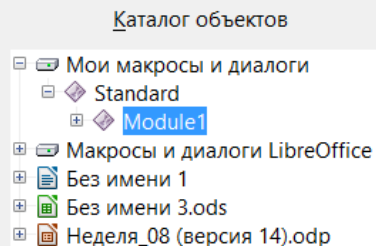
The **Макросы LibreOffice BASIC** dialog box is open, displaying the following information:

- Имя макроса** (Macro Name): `Openedu_macros_name`
- Макрос из** (Macro from): A tree view showing the macro's location: `Мои макросы` > `Standard` > `Module1`.
- Существующие макросы: Module1** (Existing macros: Module1): A list containing `Main` and `Openedu_macros_name`. The `Openedu_macros_name` macro is currently selected.

On the right side of the dialog, there are several buttons: **Выполнить** (Execute), **Закрыть** (Close), **Назначить...** (Assign...), **Правка** (Edit), **Удалить** (Delete), **Управление...** (Manage...), and **Справка** (Help).

Макросы: как исправить макрос

https://help.libreoffice.org/Basic/Programming_with_Basic



```
sub Openedu_macros_name
rem -----
rem define variables
dim document as object
dim dispatcher as object
rem -----
rem get access to the document
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")
rem -----
dim args1(4) as new com.sun.star.beans.PropertyValue
args1(0).Name = "CharFontName.StyleName"
args1(0).Value = "Полужирный"
args1(1).Name = "CharFontName.Pitch"
args1(1).Value = 2
args1(2).Name = "CharFontName.CharSet"
args1(2).Value = -1
args1(3).Name = "CharFontName.Family"
args1(3).Value = 1
args1(4).Name = "CharFontName.FamilyName"
args1(4).Value = "IzhitsaShadowC"
dispatcher.executeDispatch(document, ".uno:CharFontName", "", 0, args1())
rem -----
dim args2(2) as new com.sun.star.beans.PropertyValue
args2(0).Name = "FontHeight.Height"
args2(0).Value = 26
args2(1).Name = "FontHeight.Prop"
args2(1).Value = 100
args2(2).Name = "FontHeight.Diff"
args2(2).Value = 0
dispatcher.executeDispatch(document, ".uno:FontHeight", "", 0, args2())
rem -----
dim args3(0) as new com.sun.star.beans.PropertyValue
args3(0).Name = "Bold"
args3(0).Value = true
dispatcher.executeDispatch(document, ".uno:Bold", "", 0, args3())
end sub
```

Интересные факты о текстовых процессорах



1. **Панграмма** (с греч. «все буквы»), или разнобуквица — текст, использующий все буквы алфавита. Панграммы используются в текстовых процессорах для демонстрации шрифтов, проверки передачи текста по линиям связи, тестирования печатающих устройств.

Microsoft Windows 47 букв: «Съешь же ещё этих мягких французских булок, да выпей чаю»

GNU/Linux (Gnome) 45 букв: «В чашах юга жил бы цитрус? Да, но фальшивый экземпляр!»

Самая короткая панграмма: «Шеф взъярён тчк щипцы с эхом гудбай Жюль» (33 буквы).

2. Скрытые **незадокументированные возможности** текстовых редакторов

«**=rand(*m*, *n*)**» – если ввести этот текст без кавычек и нажать Enter, то сгенерируется псевдослучайный текст, состоящий из *m* абзацев по *n* предложений в каждом абзаце.

«**=lorem(*m*, *n*)**» – аналогично сгенерируется искажённый отрывок из философского трактата Цицерона «О пределах добра и зла», написанного на латинском языке в 45 году до Р. Х. (впервые этот текст был применен для набора шрифтовых образцов в XVI веке).

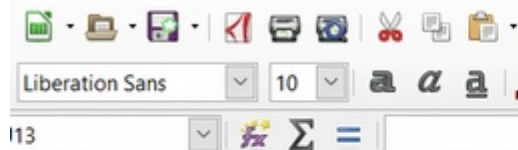
«**dt**» + **F3** — сгенерируется 1 абзац текста в LibreOffice.

Запрет на ввод некорректных значений в ячейку



Без имени 3 - LibreOffice Calc

Файл Правка Вид Вставка Формат Лист Данные Сервис Окно Справка



Оценка за экзамен по
курсу «Информатика для
вузовов»

- Сортировка...
- Сортировать по возрастанию
- Сортировать по убыванию
- Автофильтр
- Ещё фильтры
- Задать диапазон...
- Выбрать диапазон...
- Обновить диапазон
- Сводная таблица
- Содержимое ячейки
- Проверка...**
- Промежуточные итоги...
- Форма...
- Потоки...
- Источник XML...
- Совмещённые операции...
- Текст по столбцам...
- Объединить...
- Группа и структура
- Статистика

Допустимые оценки

отлично
хорошо
удовлетворительно

Проверка вводимых значений

Условия Помощь при вводе Действия при ошибке

Разрешить: **Диапазон ячеек**

☒ Пропуск пустых ячеек

☒ Показать список выделений

☐ Сортировать записи по возрастанию

Источник: \$Лист1.\$E\$4:\$E\$6

Допустимо только непрерывное выделение строк и столбцов или возвращающую диапазон ячеек или массив.

Проверка вводимых значений

Условия Помощь при вводе Действия при ошибке

☒ Показывать сообщение об ошибке при вводе недопустимых значений

Содержимое

Действие: **Стоп**

Заголовок: **Стоп**

Сообщение об ошибке: Предупреждение
Информация
Макрос

Условное форматирование

Без имени 3 - LibreOffice Calc

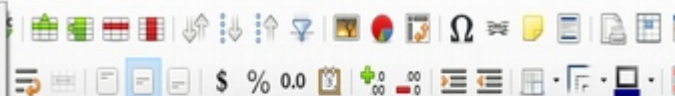
Файл Правка Вид Вставка Формат Лист Данные Сервис Окно Справка

Лiberation Sans 14

E18

	A	B	C	D
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				

- Текст
- Интервал
- Выравнивание
- Формат чисел
- Стили
- Отменить форматирование Ctrl+M
- Ячейки... Ctrl+1
- Строка
- Столбец
- Объединить ячейки
- Символы...
- Абзац...
- Страница...
- Диапазоны печати
- Условное форматирование
- Стили автоформата...
- Изображение
- Объект
- Форма
- Привязка
- Расположение
- Отражение
- Группировка



- Условие...
- Цветовая шкала...
- Гистограмма...
- Набор пиктограмм...
- Дата...
- Управление...

Успеваемость по «Русскому языку»

ФИО	Оценка
Гейтс Билл	3
Джобс Стив	2
Клейнрок Леонард	4
Ритчи Деннис	5
Столлман Ричард	5
Таненбаум Эндрю	4
Торвальдс Линус	5
Цукербергер Марк	3
Шеннон Клод	5

Условия

Условие 1

Значение ячейки меньше или равно 3

Применить стиль

- Неуспевающие
- Создать стиль...
- Базовый
- Заголовок
- Заголовок1
- Неуспевающие
- Результат
- Результат2

Добавить Удалить

Диапазон ячеек

Диапазон: D5:D13

Справка

Фильтры для заполненных таблиц

Данные Сервис Окно Справка

Сортировка...
Сортировать по возрастанию
Сортировать по убыванию
Автофильтр
Ещё фильтры
Задать диапазон...
Выбрать диапазон...
Обновить диапазон
Сводная таблица
Содержимое ячейки
Проверка...
Промежуточные итоги...
Форма...
Потоки...
Источник XML...
Совмещённые операции...
Текст по столбцам...
Объединить...
Группа и структура
Статистика

Успеваемость по «Русскому языку»

ФИО	Оценки
Гейтс Билл	3
Джобс Стив	2
Клейнрок Леонард	4
Ритчи Деннис	5
Столлман Ричард	5
Таненбаум Эндрю	4
Торвальдс Линус	5
Цукербергер Марк	3
Шеннон Клод	5

По возрастанию
По убыванию
10 первых
Пусто
Не пусто
Стандартный фильтр...
Поиск элементов...

☒ Гейтс Билл
☒ Джобс Стив
☒ Клейнрок Леонард
☒ Ритчи Деннис
☒ Столлман Ричард
☒ Таненбаум Эндрю
☒ Торвальдс Линус
☒ Все

OK Отменить

Стандартный фильтр

Условия фильтра

Операция	Имя поля	Условие	Значение
	ФИО	Начинается с	T
	- нет -	=	
	- нет -	=	
	- нет -	=	

Параметры

Справка OK Отмена

Успеваемость по «Русскому языку»

ФИО	Оценки
Таненбаум Эндрю	4
Торвальдс Линус	5

Расчёт доверительного интервала



В русской версии MS Office аналогичные функции называются

- «ОКРУГЛ»
- «ДОВЕРИТ.СТЬЮДЕНТ»
- «СТАНДОТКЛОН»
- «СЧЁТ»

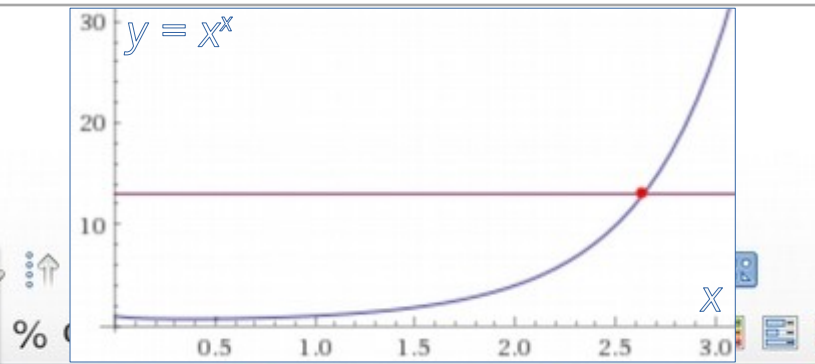
Доверительная вероятность принята равной 95% (типичное используемое учёными значение)

C16			Σ	=	=ROUND(CONFIDENCE.T(1-0,95;STDEV(C5:C13);COUNT(C5:C13));1)
	A	B	C		
1					
2					
3					
4					Время работы моей программы при расчёте на моём компьютере числа 2017!, с
5					15,7
6					14,0
7					14,3
8					15,6
9					15,4
10					15,9
11					14,4
12					15,8
13					15,3
14					
15			Среднее арифмитическое	15,2	
16			Доверительный интервал	0,60	
17			Итог	от 14,6 до 15,8	

Функция «Подбор параметра»

Решить уравнение: $x^x = 13$ (найти x).

Решение: $2^2=4$, $3^3=27 \Rightarrow 2 < x < 3$.



Подбор параметра

Предопределённые

Целевая ячейка:

Целевое значение:

Изменяемая ячейка:

OK Отменить Справка

LibreOffice Calc

Подбор параметра завершён. Результат: 2,6410619166

Вставить результат в изменяемую ячейку?

Да Нет



Без имени 3.ods - LibreOffice Calc

Файл Правка Вид Вставка Формат Лист Данные Сервис Окно Справка

Лiberation Sans 10

Б9

	A	B	C
1			
2			
3		$x^x = 13, x = ?$	
4		$y = x^x$	
5			
6			
7			
8		$x = 3$	
9		$y = 27$	
10			
11			
12			
13			
14			
15			

Сервис

- Проверка орфографии... F7
- Автопроверка орфографии Shift+F7
- Тезаурус... Ctrl+F7
- Язык
- Параметры автозамены...
- Подбор параметра...**
- Решатель...
- Зависимости
- Сценарии...
- Совместно использовать документ...
- Защитить лист...
- Защитить документ...
- ☒ Автовывод
- Макросы
- Настройка фильтров XML...
- Управление расширениями...
- Настройка...
- Параметры...

Вакансии со знанием Excel (в т.ч. макросов) на 2021 ГОД



Аналитик данных

Ароматный мир ✓

Санкт-Петербург

Планирование и расчет эффективности маркетинговых акций. Факторный анализ влияния акций с помощью регрессионных моделей и проверки гипотез. Подготовка презентаций по...

Знания Excel на уровне продвинутого пользователя (визуализация, функции (index, match и т.п.), **макросы**). Аналитический склад ума, абстрактное мышление, умение...

Аналитик

Национальный исследовательский университет Высшая школа экономики ✓

Санкт-Петербург

Сбор, обработка и проверка больших массивов данных (реестров). Расчет показателей для отчетов на основе реестров. Формирование отчетов по заданной методике.

Продвинутый пользователь Excel (включая обработку массивов данных, сводные таблицы, **макросы**, функции). Владение языками R, Python (как преимущество). Продвинутый пользователь Power...

Программист C#

АО ИнфоТеКС ✓

Санкт-Петербург

Настройка шаблона для веб-сайта с документацией: HTML, CSS, JavaScript, XML. Поддержка Word-шаблонов и **макросов** к ним на C#.

Опыт разработки веб приложений на asp.net. Опыт разработки плагинов или **макросов** к excel и word документам. Опыт разработки плагинов к...



Специалист по внедрению BPM-систем

ELMA ✓

Санкт-Петербург, ● Выборгская

Реализовывать решения на платформах ELMA. Участвовать в оценке и составлении архитектуры решения. Участвовать в тестировании решения. Устанавливать, настраивать и администрировать...

Знание MS Excel (сводные таблицы, ВПР, **макросы**). Опыт работы с реляционными СУБД (MSSQL\MySQL\PostgreSQL\Oracle), знание SQL.

Представление целых чисел в ограниченной двоичной разрядной сетке (РС) компьютера



Для хранения целой переменной в памяти компьютера используется фиксированное заранее известное число бит. Например, для хранения $a=2$ в компьютерную память будет записано следующее двоичное число, если используется 32-разрядный компьютер:

0000000000000000000000000000000010₍₂₎.

Процессор за один такт работы выполняет операцию сразу со всеми 32-мя битами:

$$\begin{array}{r} 0000000000000000000000000000000010_{(2)} \\ + \\ \hline 0000000000000000000100000000000010_{(2)} \\ \hline 0000000000000000000100000000000100_{(2)} \end{array}$$

Пусть для хранения целого неотрицательного числа в переменной a используется k бит.

$$\text{MIN}(a) = 000\dots000_{(2)} = 0,$$

$$\text{MAX}(a) = 111\dots111_{(2)} = 2^k - 1.$$

$$\begin{array}{l} 999 = 1000 - 1 = 10^3 - 1 \\ 111_{(2)} = 1000_{(2)} - 1 = 2^3 - 1 \end{array}$$

Диапазон представления целых неотрицательных чисел в k -разрядной сетке: **от 0 до 2^k-1 .**



Представление целых чисел со знаком в компьютере

В ЭВМ нет способа обозначить в двоичной СС знак «МИНУС» перед числом. Способы решения этой проблемы с примерами для 4-разрядного компьютера:

- **Специальный знаковый бит (СЗБ)**
 $+5 = 0101_2$, $-5 = 1101_2$ (первый бит означает знак числа)
- **Фиксированное смещение влево (ФСВ)**
 $-5 = 0000_2$, $-4 = 0001_2$, ..., $+10 = 1111_2$ (все числа уменьшены на 5)
- **Нега-двоичная система счисления (НДСС)**
 $-5 = 1111_{-2}$, $+5 = 0101_{-2}$ (основание СС равно «-2»)
- **Обратный/инверсный код (ОК)**
 $+5 = 0101_2$, $-5 = 1010_2$ (инвертируются все биты)
- **Дополнительный код (ДК)**
 $+5 = 0101_2$, $-5 = 1011_2$ (инвертировать все биты и прибавить 1)



Целые числа со знаком в трёхразрядном коде

Для сравнения – диапазон представления целых **неотрицательных** чисел в трёхразрядной сетке: от $000_{(2)}$ до $111_{(2)}$, т. е. от 0 до 7.

Трёхразрядный код	СЗБ	ФСВ (5)	НДСС	ОК	ДК
000	+0	-5	0	+0	0
001	1	-4	1	1	1
010	2	-3	-2	2	2
011	3	-2	-1	3	3
100	-0	-1	4	-3	-4
101	-1	0	5	-2	-3
110	-2	1	2	-1	-2
111	-3	2	3	-0	-1
Диапазон	-3..+3	-5..+2	-2..+5	-3..+3	-4..+3



Целые числа со знаком в n -разрядном компьютере

Имея n -разрядный двоичный регистр, можно закодировать 2^n разных символов. Для кодирования целых чисел без знака используется диапазон от 0 до $2^n - 1$.

Каков диапазон хранимых чисел со знаком в n -разрядном регистре?

1. Специальный знаковый бит (СЗБ):

от $-(2^{n-1} - 1)$ до $+(2^{n-1} - 1)$.

min→
max→

1	1	1	1	...	1	1	1	1
0	1	1	1	...	1	1	1	1
0	0	0	0	...	0	0	0	0
1	1	1	1	...	1	1	1	1

2. Фиксированное смещение влево (ФСВ):

от $(-S)$ до $(2^n - 1 - S)$, где S – смещение.

3. Нега-двоичная система счисления (НДСС):

чётное n : от $-(2^{n-1})/3$ до $(2^{n-1})/3$,

нечётное n : от $-(2^{n-1}-1)/3$ до $(2^{n-1}-1)/3$,

любое n : от $-(2^{n-(n \bmod 2)}-1)/3$ до $(2^{n-(n \bmod 2)}-1)/3$.

...	1	0	1	0	1	0	1	0
...	0	1	0	1	0	1	0	1

4. Обратный/инверсный код (ОК):

от $-(2^{n-1} - 1)$ до $+(2^{n-1} - 1)$.

1	0	0	0	...	0	0	0	0
0	1	1	1	...	1	1	1	1

5. Дополнительный код (ДК):

от (-2^{n-1}) до $(2^{n-1}-1)$.

1	0	0	0	...	0	0	0	0
0	1	1	1	...	1	1	1	1

Как хранится число «-2» в памяти десятиразрядного компьютера?

Решение

1 шаг: записать число «+2», используя все доступные разряды

$$2_{10} = 0000000010_2$$

2 шаг: инвертировать каждый бит полученного числа:

$$0000000010_2 \rightarrow 1111111101_2$$

3 шаг: прибавить один

$$\begin{array}{r} 1111111101_2 \\ + \underline{0000000001_2} \\ \hline 1111111110_2 \end{array}$$

4 шаг: радоваться результату: $-2_{10} = 1111111110_2$ (обратный перевод выполняется так же)

Иллюстрация эффекта $2 + (-2) = 0 \rightarrow$

$$\begin{array}{r} 0000000010_2 \\ + \underline{1111111110_2} \\ \hline 1000000000_2 \end{array}$$

– это ноль, т. к. 11-го разряда нет



$$\begin{array}{r}
 + \quad 0111_2 \\
 \quad \underline{1011_2} \\
 10010_2
 \end{array}$$

$$\begin{array}{r}
 \text{СЗБ} \\
 + \quad +7 \\
 \quad \underline{-3} \\
 +2
 \end{array}$$

$$\begin{array}{r}
 \text{НДСС} \\
 + \quad +3 \\
 \quad \underline{-9} \\
 -2
 \end{array}$$

$$\begin{array}{r}
 \text{ОК} \\
 + \quad +7 \\
 \quad \underline{-4} \\
 +2
 \end{array}$$

$$\begin{array}{r}
 \text{ДК} \\
 + \quad +7 \\
 \quad \underline{-5} \\
 +2
 \end{array}$$

Как придумали правило ДК? Почему нужно инвертировать биты и прибавлять 1?

$$x_{(2,n)} + \text{inv}(x_{(2,n)}) = \dots 11111111_{(2,n)} = 2^n - 1. \text{ Пример: } 0101_{(2,4)} + 1010_{(2,4)} = 1111_{(2,4)} = 2^4 - 1$$

$$\text{inv}(x_{(2,n)}) + 1 = 2^n - x_{(2,n)}$$

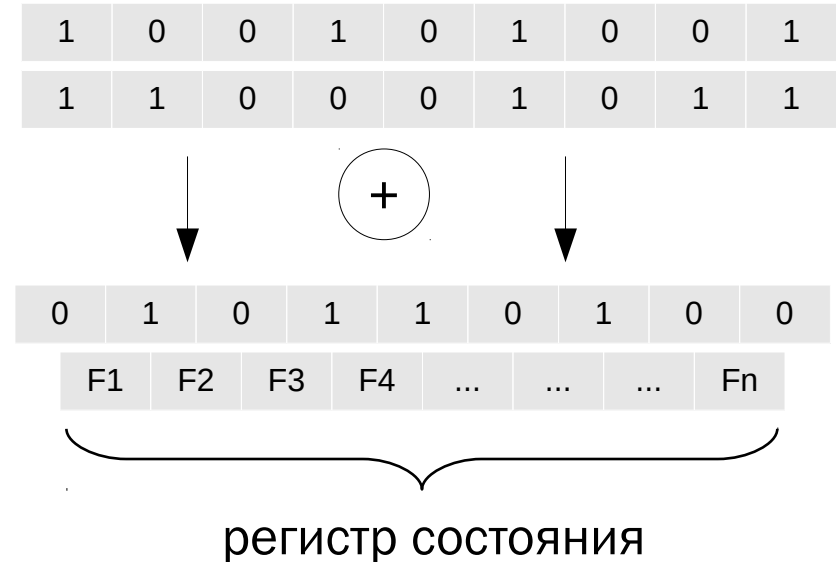
$$\text{inv}(x_{(2,n)}) + 1 = -x_{(2,n)}$$

$$a_{(2,n)} - b_{(2,n)} = a_{(2,n)} + (-b_{(2,n)}) = a_{(2,n)} + (2^n - b_{(2,n)}) = a_{(2,n)} + (\text{inv}(b_{(2,n)}) + 1)$$

Арифметические операции в ограниченной разрядной сетке



- После любой арифметической операции процессор автоматически без явной команды от программиста устанавливает флаги, характеризующие состояние процессора.
- Совокупность этих флагов называется регистром состояния.
- Программист может анализировать содержимое регистра состояния процессора для принятия решений в программе.



if (F1 == 0) then ... else ...;



SF – Sign Flag. Равен 1, если результат операции отрицателен, иначе – 0.

ZF – Zero Flag. Равен 1, если результат операции равен нулю.

PF – Parity Flag. Равен 1, если младший байт результата выполнения операции содержит чётное число единиц.

AF – Adjust Flag. Равен 1, если произошёл заём или перенос между первым и вторым полубайтом (нибблом).

CF – Carry Flag. Равен 1, если происходит перенос за пределы разрядной сетки или заём извне.

OF – Overflow Flag. Равен 1, если результат операции не помещается в разрядную сетку (при использовании дополнительного кода).



OF – Overflow Flag. Принимает значение 1, если в результате выполнения операции со знаковыми числами появляется одна из ошибок:

- 1) складываем положительные числа, получаем неположительный результат;
- 2) складываем отрицательные числа, получаем неотрицательный результат.

Примеры для 4-разрядного компьютера:

$$0100_{(2)} + 0001_{(2)} = 0101_{(2)} \text{ (CF=0, OF=0) : } +4 + 1 = +5$$

$$0110_{(2)} + 1001_{(2)} = 1111_{(2)} \text{ (CF=0, OF=0) : } +6 - 7 = -1 \text{ (1111}_2 \text{ в доп. коде это } -1_{10})$$

$$1000_{(2)} + 0001_{(2)} = 1001_{(2)} \text{ (CF=0, OF=0) : } -8 + 1 = -7$$

$$1100_{(2)} + 1100_{(2)} = 1000_{(2)} \text{ (CF=1, OF=0) : } -4 - 4 = -8$$

$$1000_{(2)} + 1000_{(2)} = 0000_{(2)} \text{ (CF=1, OF=1) : } -8 - 8 = 0$$

$$0101_{(2)} + 0100_{(2)} = 1001_{(2)} \text{ (CF=0, OF=1) : } +5 + 4 = -7$$



Пример установки флагов состояния процессора

16-разрядный компьютер

Пример 1

$$\begin{array}{rcl} 0010.0101.0000.1100 & & + 9484_{(10)} \\ + 0011.1101.1010.0100 & & +15780_{(10)} \\ \hline 0110.0010.1011.0000 & = & +25264_{(10)} \end{array}$$

CF=0, OF=0, ZF=0, AF=1, SF=0, PF=0

Пример 2

$$\begin{array}{rcl} 0110.0010.1010.1001 & & +25257_{(10)} \\ + 0011.1101.1010.1100 & & +15788_{(10)} \\ \hline 1010.0000.0101.0101 & = & -24491_{(10)} \end{array}$$

CF=0, OF=1, ZF=0, AF=1, SF=1, PF=1

Пример 3

$$\begin{array}{rcl} 1110.0111.0110.1000 & & - 6296_{(10)} \\ + 0110.0010.1011.0000 & & +25264_{(10)} \\ 1.0100.1010.0001.1000 & = & +18968_{(10)} \end{array}$$

CF=1, OF=0, ZF=0, AF=0, SF=0, PF=1