***4 ЛАБА***

1)Markdown- облегченный язык разметки(Минимум функций, Исходный текст на таком языке читается с такой же лёгкостью, как и готовый документ)

Markup-Облегченный язык разметки(предназначенные для простого и быстрого добавления форматирования в [текстовые данные](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B5_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5))

Разница в том, что markdown-разновидность markup. В markup так же входят [BBCode](https://ru.wikipedia.org/wiki/BBCode), [Textile](https://ru.wikipedia.org/wiki/Textile) и Вики-разметка.

2) **Protocol Buffers** — [протокол](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB_%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) [сериализации](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B8%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F" \o "Сериализация) (передачи) структурированных данных, предложенный [Google](https://ru.wikipedia.org/wiki/Google_(%D0%BA%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D0%B8%D1%8F)) как эффективная бинарная альтернатива текстовому формату [XML](https://ru.wikipedia.org/wiki/XML). Разработчики сообщают, что Protocol Buffers проще, компактнее и быстрее, чем [XML](https://ru.wikipedia.org/wiki/XML), поскольку осуществляется передача бинарных данных, оптимизированных под минимальный размер сообщения. По замыслу разработчиков, сначала должна быть описана структура данных, которая затем компилируется в классы. Вместе с классами идёт код их [сериализации](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B8%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F" \o "Сериализация) в компактном формате представления. Чтение и запись данных доступна в высокоуровневых языках программирования — таких как [Java](https://ru.wikipedia.org/wiki/Java), [C++](https://ru.wikipedia.org/wiki/C%2B%2B) или [Python](https://ru.wikipedia.org/wiki/Python).

3) **CSV**  — [текстовый](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB) [формат](https://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D1%82_%D1%84%D0%B0%D0%B9%D0%BB%D0%B0), предназначенный для представления табличных данных. Строка таблицы соответствует строке текста, которая содержит одно или несколько полей, разделенных запятыми.

**TSV** — [текстовый формат](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82) для представления таблиц баз данных. Каждая запись в таблице — это строка текстового файла. Каждое поле записи отделяется от других с помощью символа [табуляции](https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D0%B1%D1%83%D0%BB%D1%8F%D1%86%D0%B8%D1%8F), точнее горизонтальной табуляции.

4) -----

5) < и > -это теги, обозначающие границы элементов.

6) **Сериализация** — процесс перевода [структуры данных](https://ru.wikipedia.org/w/index.php?title=%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1) в последовательность [байтов](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B9%D1%82). Обратной к операции сериализации является операция *десериализации* (структуризации) — создание структуры данных из битовой последовательности.

Сериализация используется для передачи [объектов](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) по сети и для сохранения их в [файлы](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB).

7) Комменты в YAML- #

8)заголовки: # -1 уровень; ## -2 уровень и тд

Блоки кода разделяются с помощью \*\*\* и ---

Чтобы отметить фрагмент строки, содержащий код, необходимо окружить его обратными апострофами «`»

\*\*да\*\* = **да**

\*да\* = *да(курсив)*

~~да~~ = ~~да~~

9) Viber: JSON; WhatsApp: JSON; Telegram: TL(Type Languare, собственный ML телеграма)/JSON; VK: JSON; Twitter: JSON

10) SVG (Scalable **Vector** Graphics — масштабируемая векторная графика) — язык разметки масштабируемой векторной графики

11) **< a href="http://www.seoded.ru/">Ссылка на сайт< /a>**

12) JSON-текст представляет собой (в закодированном виде) одну из двух структур:

* Набор пар *ключ: значение*. В различных языках это реализовано как [запись](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BF%D0%B8%D1%81%D1%8C_(%D1%82%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)), [структура](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), [словарь](https://ru.wikipedia.org/wiki/%D0%90%D1%81%D1%81%D0%BE%D1%86%D0%B8%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2), [хеш-таблица](https://ru.wikipedia.org/wiki/%D0%A5%D0%B5%D1%88-%D1%82%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D0%B0), [список](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) с ключом или [ассоциативный массив](https://ru.wikipedia.org/wiki/%D0%90%D1%81%D1%81%D0%BE%D1%86%D0%B8%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2). Ключом может быть только строка ([регистрозависимость](https://ru.wikipedia.org/wiki/%D0%A7%D1%83%D0%B2%D1%81%D1%82%D0%B2%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D1%8C_%D0%BA_%D1%80%D0%B5%D0%B3%D0%B8%D1%81%D1%82%D1%80%D1%83_%D1%81%D0%B8%D0%BC%D0%B2%D0%BE%D0%BB%D0%BE%D0%B2" \o "Чувствительность к регистру символов) не регулируется стандартом, это остаётся на усмотрение программного обеспечения. Как правило, регистр учитывается программами — имена с буквами в разных регистрах считаются разными, например[[5]](https://ru.wikipedia.org/wiki/JSON#cite_note-5)), значением — любая форма. Повторяющиеся имена ключей допустимы, но не рекомендуются стандартом; обработка таких ситуаций происходит на усмотрение программного обеспечения, возможные варианты — учитывать только первый такой ключ, учитывать только последний такой ключ, генерировать ошибку.
* Упорядоченный набор *значений*. Во многих языках это реализовано как [массив](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [вектор](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D0%B4%D0%B5%D0%BA%D1%81%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2), список или [последовательность](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D1%8C).

В YAML:

lesson1: # объект

Time:#ключ 10:00-11:30 #значение

В XML:

<корневой>

<потомок>

<подпотомок>.....</подпотомок>

</потомок>

</корневой>

***5 ЛАБА***

CF — флаг переноса. Устанавливается в 1, если результат предыдущей операции не уместился в приемнике и произошел перенос из старшего бита или если требуется заем (при вычитании), иначе устанавливается в 0. Например, после сложения слова 0FFFFh и 1, если регистр, в который надо поместить результат, — слово, в него будет записано 0000h и флаг CF = 1.  
PF — флаг четности. Устанавливается в 1, если младший байт результата предыдущей команды содержит четное число бит, равных 1; устанавливается в 0, если число единичных бит нечетное. (Это не то же самое, что делимость на два. Число делится на два без остатка, если его самый младший бит равен нулю, и не делится, если он равен 1.)  
AF — флаг полупереноса или вспомогательного переноса. Устанавливается в 1, если в результате предыдущей операции произошел перенос (или заем) из третьего бита в четвертый. Этот флаг используется автоматически командами двоично-десятичной коррекции.  
ZF — флаг нуля. Устанавливается в 1, если результат предыдущей команды — ноль.  
SF — флаг знака. Этот флаг всегда равен старшему биту результата.

OF — флаг переполнения. Этот флаг устанавливается в 1, если результат предыдущей арифметической операции над числами со знаком выходит за допустимые для них пределы. Например, если при сложении двух положительных чисел получается число со старшим битом, равным единице (то есть отрицательное) и наоборот.

1. Для 2-чного 9-разрядного числа ОДЗ=(0;511)=(0;29-1)
2. Формула 2n-1(где n-кол-во разрядов)
3. 1100000
4. PF=0; SF=1;
5. CF=OF=0, если результат уместился в приемник и не вышел за допустимые значения;

CF=OF=1, если из-за переноса старшего бита результат вышел за допустимые пределы.

1. ZF=0, если складывают 0 и 0 или противоположные числа.

7)1 способ(дополнение до единицы): если число положительное, то в старший разряд (который является знаковым) записывается ноль, а далее записывается само число;

если число отрицательное, то код получается инвертированием представления модуля числа (получается обратный код);

если число является нулем, то его можно представить двумя способами: +0+0 (000…000)(000…000) или −0−0 (111…111)(111…111).

Пример: переведём число −13−13 в двоичный восьмибитный код. Прямой код модуля −13−13: 0000110100001101, инвертируем и получаем 1111001011110010. Для получения из дополнительного кода самого числа достаточно инвертировать все разряды кода.

2 способ(дополнение до двух): если число неотрицательное, то в старший разряд записывается ноль, далее записывается само число;

если число отрицательное, то все биты модуля числа инвертируются, то есть все единицы меняются на нули, а нули — на единицы, к инвертированному числу прибавляется единица, далее к результату дописывается знаковый разряд, равный единице.

В качестве примера переведём число −5−5 в дополнительный восьмибитный код. Прямой код модуля −5−5: 00001010000101, обратный — 11110101111010, прибавляем 11, получаем 11110111111011, приписываем 11 в качестве знакового разряда, в результате получаем 1111101111111011.

8)OpenDocument Format (ODF, сокращённое от OASIS Open Document Format for Office Application — открытый формат документов для офисных приложений) — открытый формат файлов документов для хранения и обмена редактируемыми офисными документами, в том числе текстовыми документами (такими как заметки, отчёты и книги), электронными таблицами, рисунками, базами данных, презентациями. Он дает доступ к содержанию документов независимо от вида и версии приложения, в котором они были созданы. ODF претендует на то, чтобы быть более простым, более удобным, более понятным форматом, чем ставшие стандартом де-факто форматы от "Майкрософт" rtf, doc, xls и т.д.

9) Выделите один или несколько столбцов и нажмите клавишу CTRL, чтобы выделить другие несмежные столбцы.

Щелкните выделенные столбцы правой кнопкой мыши и выберите команду **Скрыть**.

10) правила: больше, меньше, между, равно, текст содержит, дата, повторяющиеся значения.

11) Макросы можно использовать для написания хакерских программ и вирусов.

Макросы можно использовать не только в текстовых процессорах, но и в электронных таблицах.

Офисные пакеты предоставляют возможность отслеживать действие пользователя и записывать их в виде программного кода в макросе.

В LibreOffice макросы можно писать не только на языке Basic, но также на Python.

12)