

3 Проверка корректности

1 Задание

Пользуясь полученными в л. р. № 1 знаниями и результатами, достигнутыми при выполнении л. р. № 2, написать программу, проверяющую корректность синтаксиса входного потока в соответствии с вариантом.

При обнаружении ошибок программа должна сообщать о них в понятной пользователю форме. Программа должна завершаться с кодом 0, если завершилась успешно, и с отличным от 0 кодом в случае ошибки.

В тех случаях, где ввод предусмотрен исключительно из стандартного файла, необходимо реализовать интерактивный режим работы программы.

2 Варианты

1. Подмножество языка Python: целый и строковый типы, списки, словари, операции над ними и команда del.
2. Конфигурационные файлы формата INI.
3. Файлы паролей /etc/passwd.
4. Подмножество языка Prolog.
5. Файлы описания DNS-зон сервера BIND.
6. Репозитории APT.
7. Предложение SELECT языка SQL.
8. Диалект языка Lisp с возможностью определения лямбда- и обычных функций.
9. Конфигурационный файл команды sudo /etc/sudoers.
10. Файлы конфигурации xorg.conf сервера X-org.
11. Подмножество языка Shell с инструкциями ветвления, циклов и создания функций.
12. Предложение INSERT языка SQL.
13. Предложение UPDATE языка SQL.
14. Предложение DELETE языка SQL.

15. Подмножество языка C++: типы `int`, `std::string`, `std::list<>`, `std::map<>` и стандартные операции над ними.
16. Таблицы формата CSV.
17. Сценарии программируемого инженерного калькулятора с базовыми арифметическими операциями, операцией возведения в степень, числами в форматах с десятичной запятой и экспоненциальном, тригонометрическими и логарифмическими функциями.
18. Файлы Cascading Style Sheets (CSS).
19. Булевы уравнения.
20. Алгебраические выражения.
21. Файлы электронной почты.

```
calc> a = 5
      ОК
calc> a 5 =
      ^ syntax error: operator expected
calc> 1 + a / (p - pi)
      ОК
calc> const pi
      ^ syntax error: '=' expected
```