ЭТАП 2

1. **Необходимо реализовать консольную программу, которая бы фильтровала поток текстовой информации, подаваемой на вход, и на выходе показывала лишь те строчки, которые содержат слово, переданное программе на вход в качестве аргумента (ввод всех строк осуществляется последовательно, одна за другой, окончание ввода обозначается вводом пустой строки).**
   1. В аргументах может быть передано не одно слово, а несколько (тогда должны вернуться строки, которые содержат хотя бы одно из введённых слов);
   2. В качестве аргумента может быть задано не конкретное слово, а регулярное выражение (используется для проверки КАЖДОГО СЛОВА ОТДЕЛЬНО).

|  |  |
| --- | --- |
| Примеры ввода | Примеры вывода |
| Параметр программы: abcd Строки:  abcdf;  abcd abc;  bcd adbc abc;  ghij abcd acdf;  ab ac ad; | Вывод: **abcd** abc;  ghij **abcd** acdf; |
| Параметры программы: abcd abc  Строки:  abcdf;  abcd abc;  bcd adbc abc;  ghij abcd acdf;  ab ac ad; | Вывод:  **abcd** **abc**;  bcd adbc **abc**;  ghij **abcd** acdf; |
| Параметры программы: ^ab.+  Строки:  abcdf;  abcd abc;  bcd adbc abc;  ghij abcd acdf;  ab ac ad; | Вывод:  **abcdf**;  abcd **abc**;  bcd adbc **abc**;  ghij **abcd** acdf; |
| Параметры программы: pqrst  Строки:  abcdf;  abcd abc;  bcd adbc abc;  ghij abcd acdf;  ab ac ad; | Вывод: |

*Комментарии к задаче 1:*

* 1. Регулярное выражение, являющееся невалидным (например, `\*`), должно определятся как обычная строка.

1. **Необходимо реализовать консольное приложение-калькулятор с поддержкой функционала поиска суммы/ разности/ произведения/ частного/ возведения в целую степень. Выводить корректное сообщения об ошибке в случае невалидных параметров (например, деление на 0). Реализовать поддержку нескольких операций одновременно и приоритета операций с учётом скобок ( () >> ^ >> \*/ >> ±)**

|  |  |
| --- | --- |
| Примеры ввода | Примеры вывода |
| 2 + 2 \* 2 | 6 |
| (4 + 3) \* 2 ^ -2 | 1.75 |
| 5 + 1/0 | Деление на 0 (или любое другое текстовое сообщение об ошибке) |
| (17 ^ 4 + 5 \* 974 ^ 33 + 2.24 \* 4.75)^0 | 1 |
| 4 2 \* 3 | Невалидные данные |
| 4a \* 5 | Невалидные данные |

*Комментарии к задаче 2:*

1) Рекомендуется использовать классы java.math.BigDecimal и java.math.BigInteger для поддержки операций над достаточно большими числами;

2) Разделителем целой и дробной части чисел во вводимой строке может выступать как точка, так и запятая;

3) Наличие пробелов между числами и знаками не является невалидным, в то время как наличие пробелов в самих числах — является.