Лабораторную работу я разбил на 3 части по очевидным причинам, а именно на решение польской записи (постфиксной), решение обычного выражения (обычная запись) и решение обычного выражения, прочитанного из файла. Также были несколько малых задач, таких как реализация флагов в консоли при запуске программы, создание читаемого кода.

Польскую запись я реализовал сразу. Сначала я считывал всю строку, что вводил пользователь, а после циклом разбивал её на части (сеператором выступал пробел) и каждый элемент записывал в глобальный массив line. В итоге получили массив из чисел и операций. После вызвал функцию подсчета. Я создал функцию, т к решать выражение рекурсией показалось хорошей идеей. Функция работает очень просто: мы пробегаемся с начала по нашему глобальному массиву и ищем операцию. При её нахождении мы её выполняем, результат записываем в ячейку операции. Числа, над которыми совершилась операция, удаляются, и в их ячейки вводятся пустые строчки. Выходом из рекурсии является условие того, что в массиве остался только 1 непустой элемент, и это будет ответ на данное выражение.

Чтобы программа могла решать выражения, где 3 числа стоят подряд, была создана функция remake, которая сдвигает все непустые ячейки влево.

После этого была решена задача решения обычного выражения, введённая в консоль. Было создано 2 вида стеков (2 класса) для знаковых и числовых значений, а также несколько простых логических функций. Можно было обойтись и без них, но так выглядит красивее. Сначала я так же получаю строку, введённую пользователем, и запускаю функцию. На этот раз я не пользовался рекурсией. Программа пробегает и удаляет все пробелы, после чего мы запускаем цикл, в котором мы пробегаем по полученной строке. Когда видим число, мы кидаем его в числовой стек, если операцию – в знаковый. Если у нас операция +/-, а до неё была любая другая операция, мы её (другую операцию) выполняем. Если скобка – нет. Именно поэтому к введённой строке в начале я добавляю +0. Это чтобы последняя операция была выполнена. Скобки я реализовал следующим образом: если попадается открывающая скобка, то мы её кидаем в знаковый стек, если закрывающая скобка – запускаем цикл в обратную сторону по знаковому стеку и идём пока не находим открывающую скобку, по ходу выполняя операции.

Третью главную задачу я решил просто: сделал просто чтение из файла (путь которого ввёл пользователь при запуске) строки и запустил уже сделанную функцию для решения обычного выражения.

Работа с флагами не вызвала никаких трудностей, на лекциях всё хорошо объяснили + в интернете много полезной информации. Argv[1] = наш флаг, а argv[0] путь к файлу программы. Argv[2] использовался только при работе с файлом, в нём хранился путь к файлу со строкой.

В итоге, обдумывая проделанную работу, я понял, что написать решение обычного выражения можно было гораздо легче, а именно просто преобразовать её в польскую запись (постфиксную) и запустить уже готовую функцию. Таким образом решения всей лабораторной работы свелось бы к решению 1 задачи.