Описание методов программы

Пользовательские методы

1. **main(int argc, char** argv)\*\*  
   Точка входа в программу. Проверяет наличие аргументов, инициализирует настройки, парсит аргументы и обрабатывает файлы.
2. **initialize\_options(void)**  
   Инициализирует структуру с настройками программы значениями по умолчанию.
3. **parse\_arguments(int argc, char** argv, char\* search\_pattern)\*\*  
   Обрабатывает аргументы командной строки, устанавливает соответствующие флаги в структуре options и заполняет шаблон поиска.
4. **process\_files(int argc, char** argv, const char\* search\_pattern)\*\*  
   Открывает файлы для обработки и вызывает функцию поиска для каждого файла.
5. **search\_in\_file(char** argv, const char\* pattern, FILE\* file)\*\*  
   Ищет шаблон в файле с использованием регулярных выражений, подсчитывает совпадения и выводит результаты.
6. **print\_matching\_line(char** argv, const char\* line, int match\_result, regmatch\_t match[], regex\_t\* regex, int line\_number)\*\*  
   Выводит строку с совпадением в соответствии с установленными флагами.
7. *print\_matches\_only(const char line, int match\_result, regmatch\_t match[], regex\_t* regex, char\*\* argv, int line\_number)\*\*  
   Выводит только совпадающие части строки (для флага -o).
8. **print\_line\_header(char** argv, int line\_number)\*\*  
   Выводит заголовок строки (имя файла и номер строки) в соответствии с флагами.
9. **print\_file\_summary(char** argv, int match\_count)\*\*  
   Выводит сводную информацию по файлу (для флагов -c и -l).
10. *ensure\_proper\_newline(const char line)*\*  
    Проверяет и добавляет перевод строки при необходимости.
11. **create\_regex\_flags(bool ignore\_case)**  
    Создает флаги для компиляции регулярного выражения на основе параметров.
12. *handle\_extended\_pattern(int pattern\_count, char* search\_string)\*\*  
    Обрабатывает шаблон из аргумента -e.
13. *handle\_pattern\_from\_file(int pattern\_count, char* search\_string)\*\*  
    Обрабатывает шаблоны из файла (флаг -f).
14. **handle\_default\_pattern(char** argv, char\* search\_string)\*\*  
    Обрабатывает шаблон по умолчанию (без флагов -e/-f).
15. *remove\_trailing\_newline(char line)*\*  
    Удаляет символ новой строки в конце строки.
16. *get\_last\_character(const char line)*\*  
    Возвращает последний символ строки.
17. *add\_pattern\_separator(int pattern\_count, char* pattern)\*\*  
    Добавляет разделитель шаблонов (|) при необходимости.

Системные методы, используемые в коде

1. **fprintf** - вывод форматированных данных в поток
2. **getopt\_long** - парсинг аргументов командной строки
3. **fopen/fclose** - открытие/закрытие файлов
4. **fgets** - чтение строки из файла
5. **regcomp/regexec/regfree** - работа с регулярными выражениями
6. **strcat/strcpy/strlen** - операции со строками
7. **printf** - вывод данных в стандартный поток вывода

Алгоритм работы программы

1. **Инициализация**
   * Программа запускается с аргументами командной строки
   * Проверяется наличие минимального количества аргументов
   * Инициализируются настройки по умолчанию
2. **Парсинг аргументов**
   * Обрабатываются флаги командной строки (-i, -v, -c, -l и др.)
   * Формируется шаблон поиска (из аргументов -e, -f или первого позиционного аргумента)
   * Устанавливаются соответствующие флаги в структуре options
3. **Обработка файлов**
   * Определяются файлы для обработки (из оставшихся аргументов командной строки)
   * Для каждого файла:
     + Открывается файл
     + Компилируется регулярное выражение на основе шаблона
     + Построчно проверяется файл на соответствие регулярному выражению
     + В зависимости от флагов выводятся:
       - Совпадающие строки (или их части)
       - Номера строк
       - Имена файлов
       - Количество совпадений
     + Освобождаются ресурсы регулярного выражения
     + Закрывается файл
4. **Вывод результатов**
   * В зависимости от флагов выводятся:
     + Только имена файлов с совпадениями (-l)
     + Количество совпадений (-c)
     + Подробная информация о каждом совпадении (по умолчанию)
     + Только совпадающие части (-o)