



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе №6 по дисциплине «Операционные системы»

Тема Системный вызов open

Студент Романов А.В.

Группа ИУ7-63Б

Оценка (баллы) _____

Преподаватель Рязанова Н. Ю.

1 | Используемые структуры

Листинг 1.1: Листинг структуры filename

```
1 struct filename {  
2     const char    *name; /* pointer to actual string */  
3     const __user char *uptr; /* original userland pointer */  
4     int          refcnt;  
5     struct audit_names *aname;  
6     const char    iname[];  
7 };
```

Листинг 1.2: Листинг структуры open_flags

```
1 struct open_flags {  
2     int open_flag;  
3     umode_t mode;  
4     int acc_mode;  
5     int intent;  
6     int lookup_flags;  
7 };
```

Листинг 1.3: Листинг структуры nameidata

```
1  
2 #define EMBEDDED_LEVELS 2  
3 struct nameidata {  
4     struct path path;  
5     struct qstr last;  
6     struct path root;  
7     struct inode *inode; /* path.dentry.d_inode */  
8     unsigned int flags;  
9     unsigned seq, m_seq, r_seq;  
10    int last_type;  
11    unsigned depth;  
12    int total_link_count;  
13    struct saved {  
14        struct path link;  
15        struct delayed_call done;  
16        const char *name;
```

```

17     unsigned seq;
18 } *stack, internal[EMBEDDED_LEVELS];
19 struct filename *name;
20 struct nameidata *saved;
21 unsigned root_seq;
22 int dfd;
23 kuid_t dir_uid;
24 umode_t dir_mode;
25 } __randomize_layout;

```

Флаги системного вызова open()

O_EXEC — открыть только для выполнения (результат не определен, при открытии директории).

O_RDONLY — открыть только на чтение.

O_RDWR — открыть на чтение и запись.

O_SEARCH — открыть директорию только для поиска (результат не определен, при использовании с файлами, не являющимися директорией).

O_WRONLY — открыть только на запись.

O_APPEND — файл открывается в режиме добавления, перед каждой операцией записи файловый указатель будет устанавливаться в конец файла.

O_CLOEXEC — включает флаг `close-on-exec` для нового файлового дескриптора, указание этого флага позволяет программе избегать дополнительных операций `fcntl F_SETFD` для установки флага **FD_CLOEXEC**.

O_CREAT — если файл не существует, то он будет создан.

O_DIRECTORY — если файл не является каталогом, то `open` вернёт ошибку.

O_DSYNC — файл открывается в режиме синхронного ввода-вывода (все операции записи для соответствующего дескриптора файла блокируют вызывающий процесс до тех пор, пока данные не будут физически записаны).

O_EXCL — если используется совместно с **O_CREAT**, то при наличии уже созданного файла вызов завершится ошибкой.

O_NOCTTY — если файл указывает на терминальное устройство, то оно не станет терминалом управления процесса, даже при его отсутствии.

O_NOFOLLOW — если файл является символической ссылкой, то `open` вернёт ошибку.

O_NONBLOCK — файл открывается, по возможности, в режиме `non-blocking`, то есть никакие последующие операции над дескриптором файла не заставляют в дальнейшем вызывающий процесс ждать.

O_RSYNC — операции записи должны выполняться на том же уровне, что и **O_SYNC**.

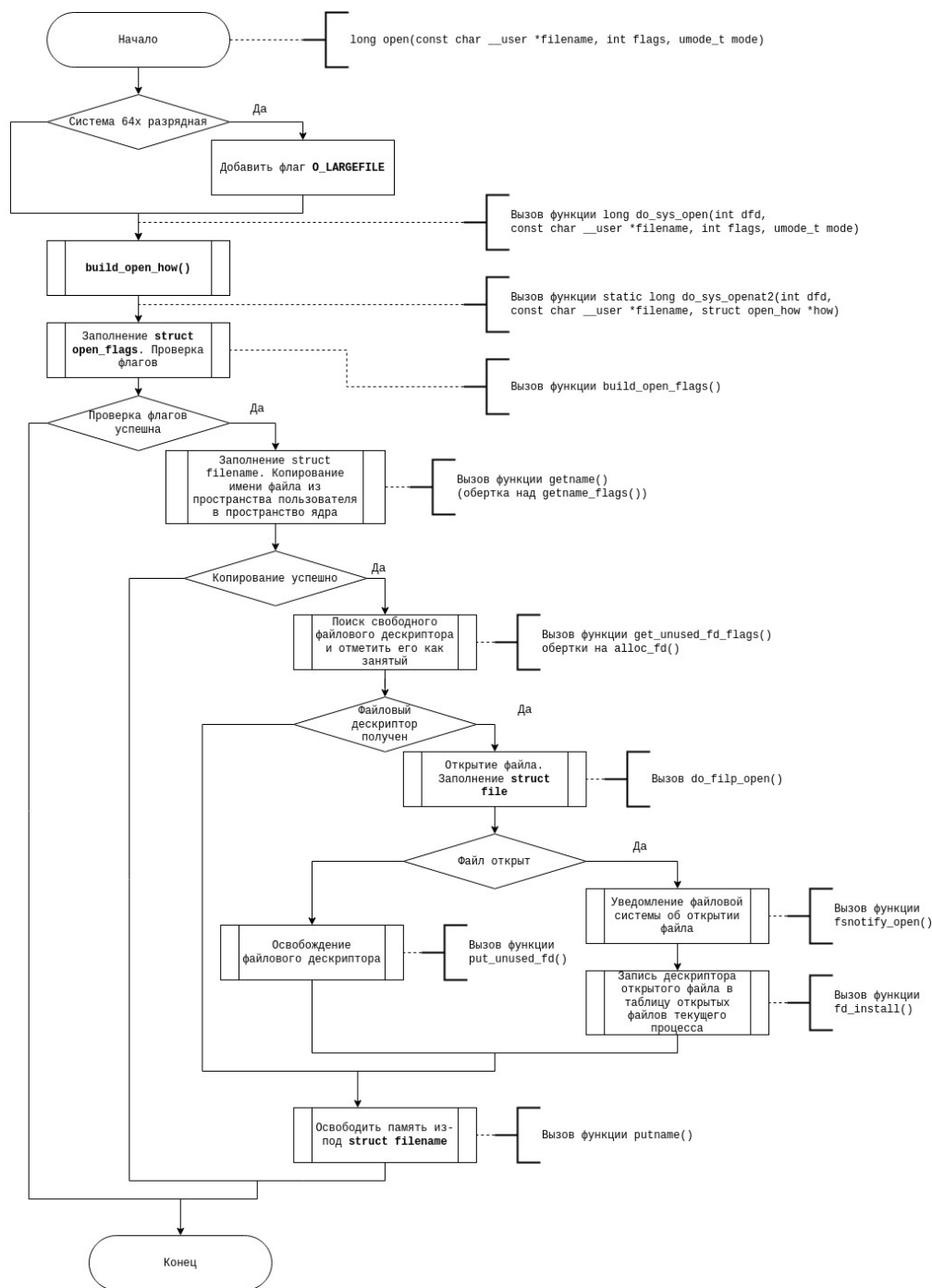
O_SYNC — файл открывается в режиме синхронного ввода-вывода (все операции записи для соответствующего дескриптора файла блокируют вызывающий процесс до тех пор, пока данные не будут физически записаны).

O_TRUNC — если файл уже существует, он является обычным файлом и заданный режим позволяет записывать в этот файл, то его длина будет урезана до нуля.

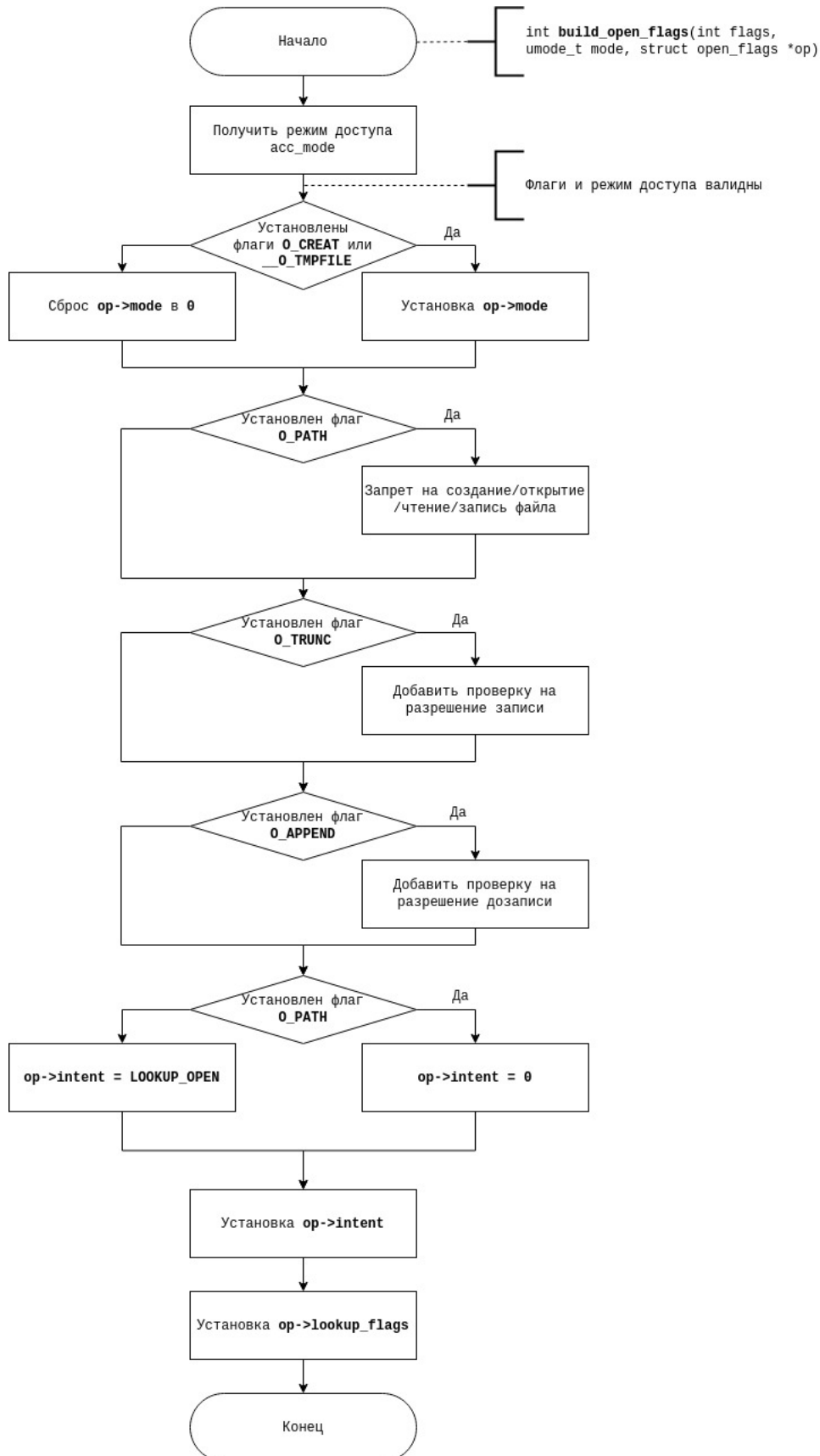
O_LARGEFILE — позволяет открывать файлы, размер которых не может быть представлен типом `off_t` (`long`).

O_TMPFILE — при наличии данного флага создаётся неименованный временный файл.

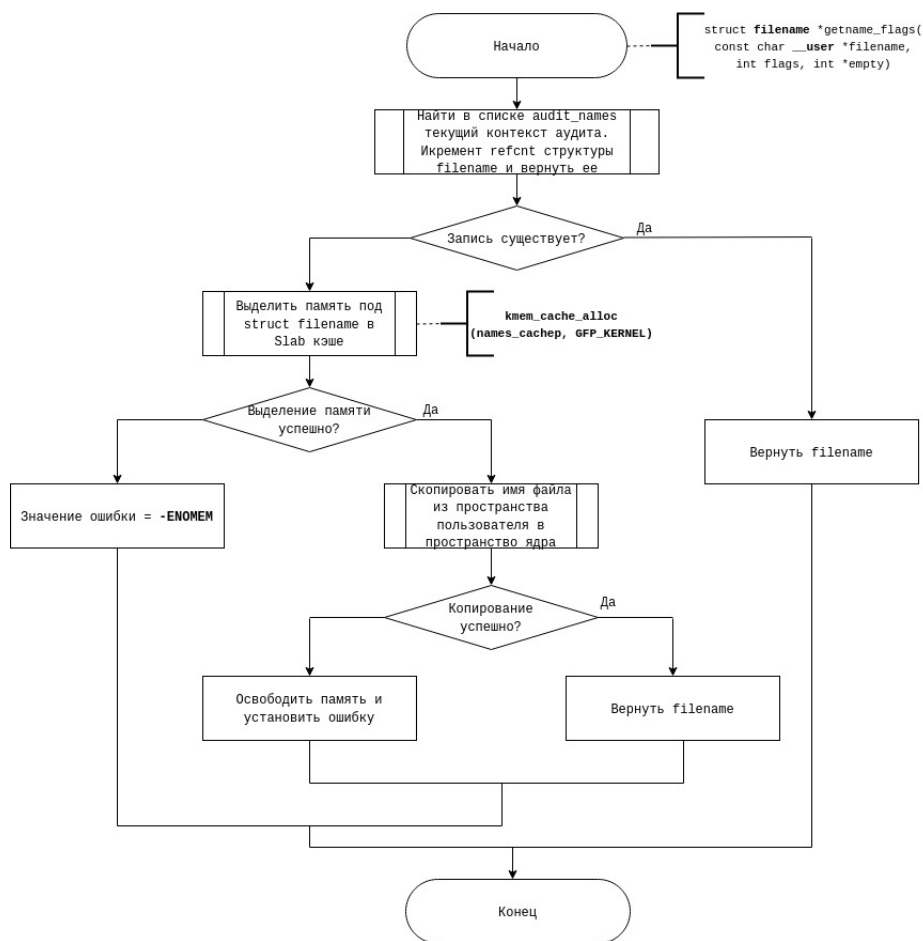
2 | Схемы алгоритмов



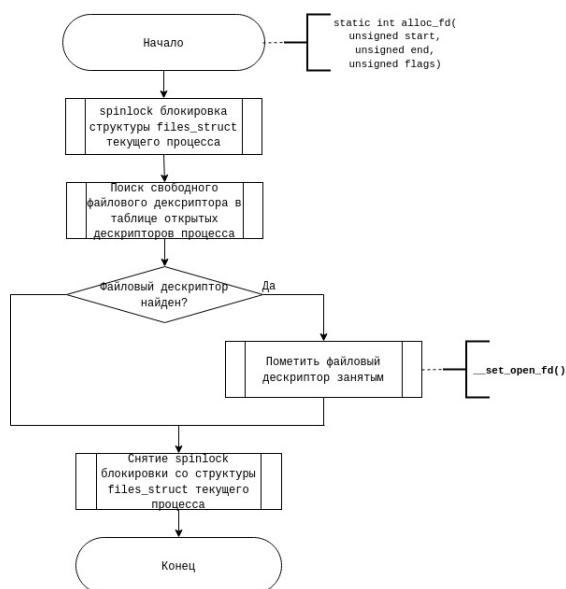
2.1 Схема работы алгоритма функции build_open_flags



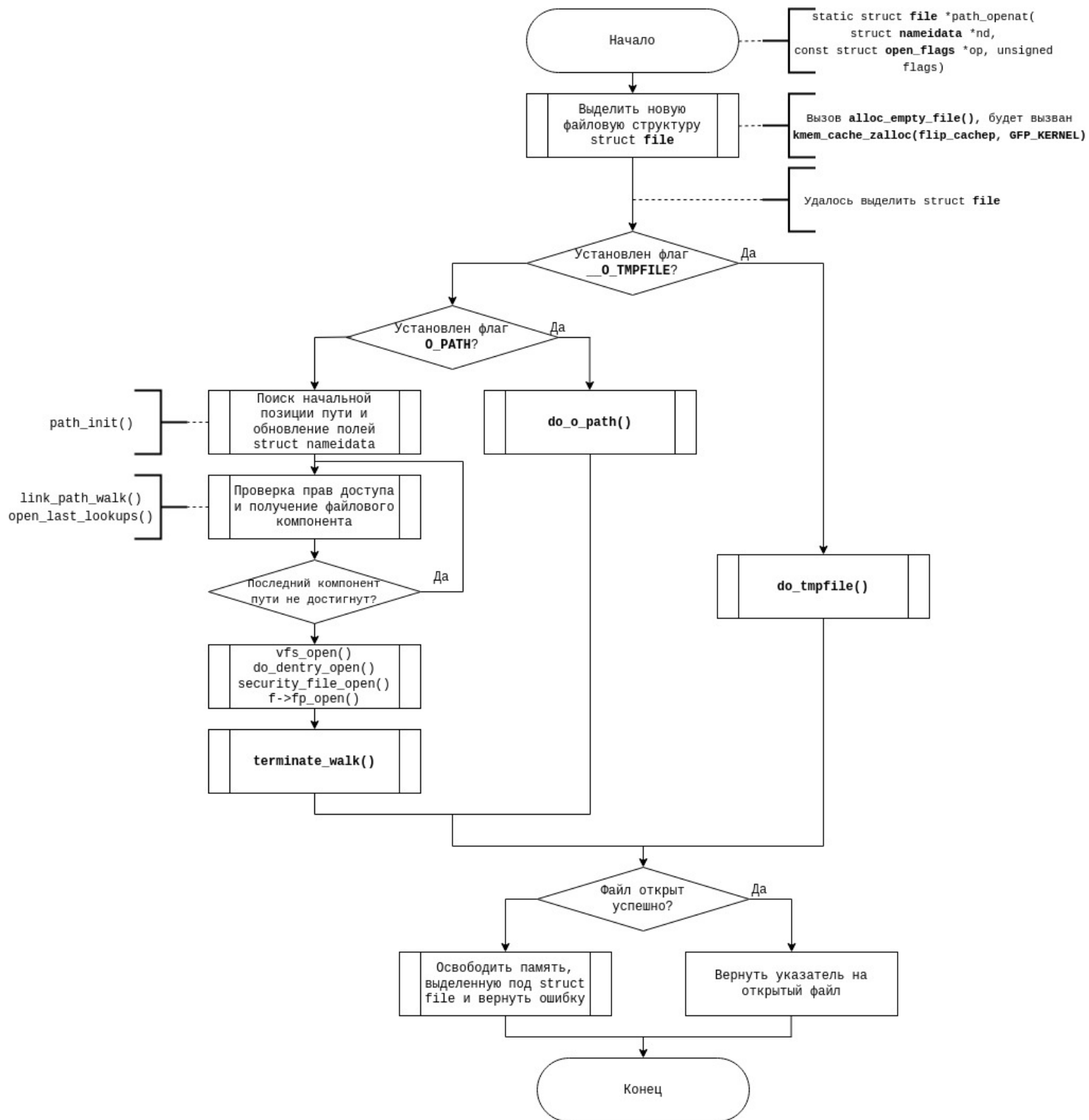
2.2 Схема работы алгоритма функции getname_flags



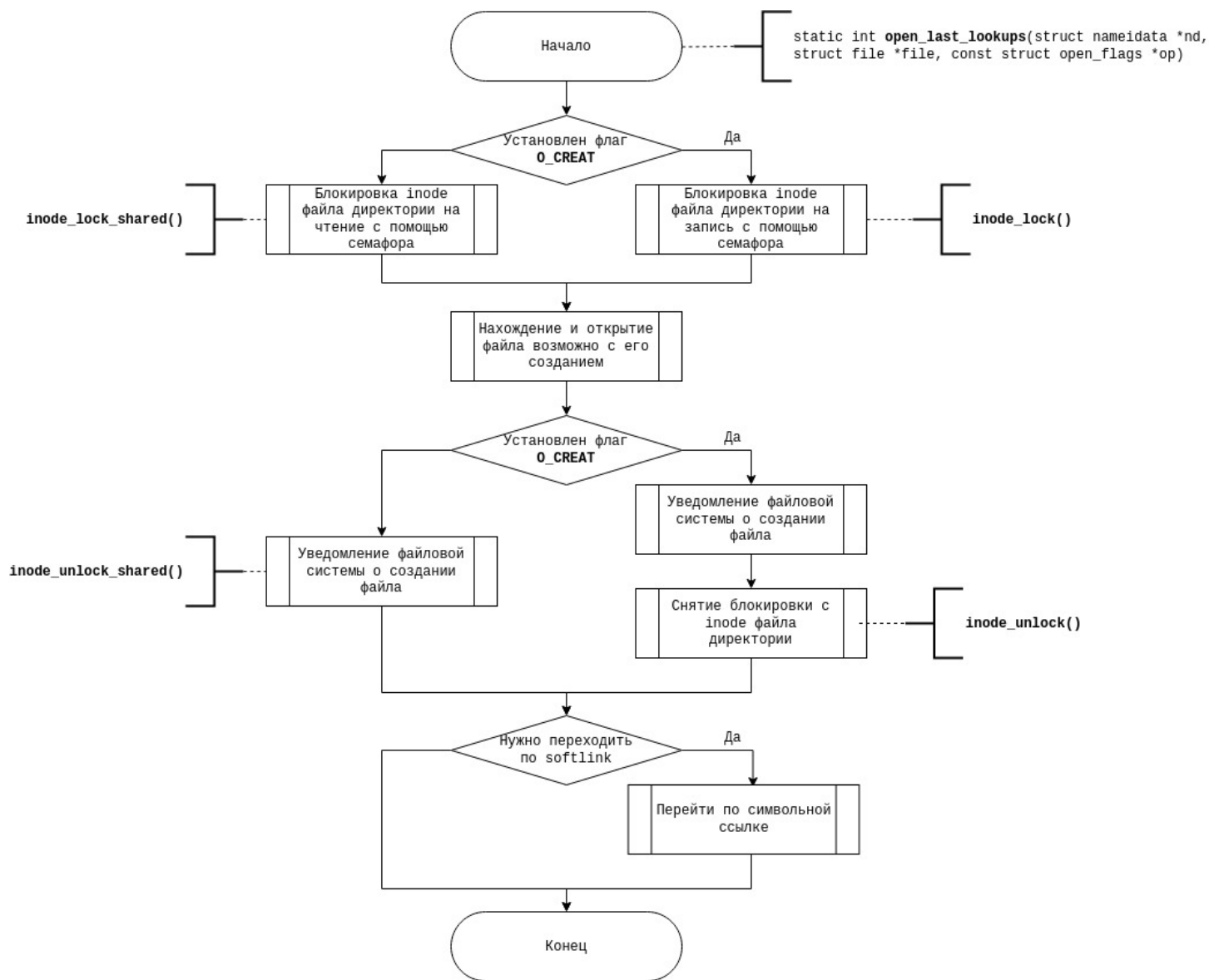
2.3 Схема работы алгоритма функции alloc_fd



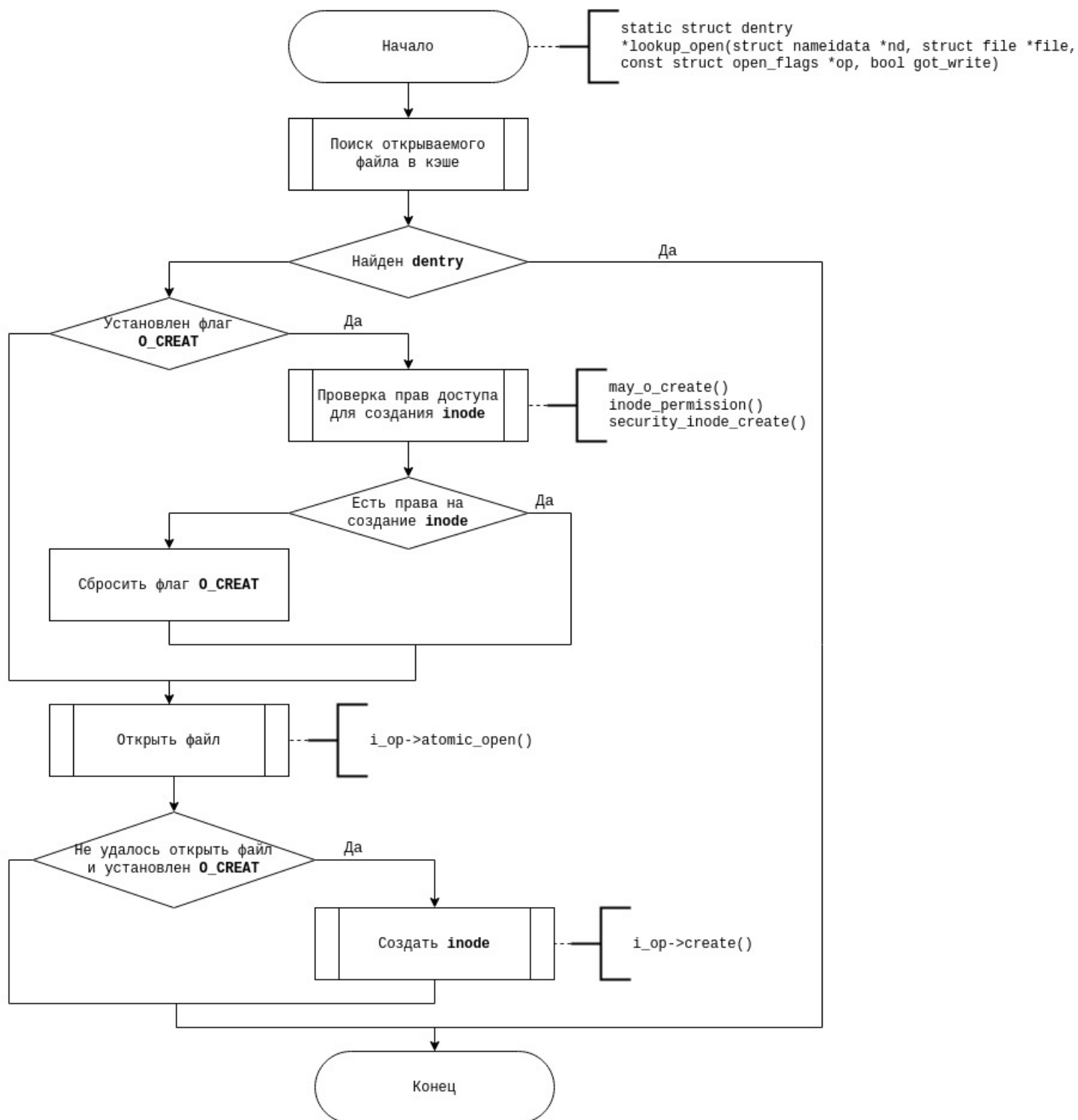
2.4 Схема работы алгоритма функции path_openat



2.5 Схема работы алгоритма функции open_last_lookups



2.6 Схема работы алгоритма функции lookup_open



2.7 Схема работы алгоритма функции set_nameidata

