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 №5 по курсу "Операционные системы"

Тема Системный вызов open

Студент Калашков П. А.

Группа ИУ7-66Б

Оценка (баллы)

Преподаватели Рязанова Н. Ю.

Москва — 2023 г.

Используемые структуры

Листинг 1 – Структура open_flags

```
struct open_flags {
        int open_flag;
        umode_t mode;
        int acc_mode;
        int intent;
        int lookup_flags;
};
```

Листинг 2 – Структура filename

```
struct filename {
        const char              *name;  /* pointer to actual string */
        const __user char       *uptr;  /* original userland pointer */
        int                     refcnt;
        struct audit_names      *aname;
        const char              iname[];
};
```

Листинг 3 – Структура nameidata

```
struct nameidata {
        struct path     path;
        struct qstr     last;
        struct path     root;
        struct inode    *inode; /* path.dentry.d_inode */
        unsigned int    flags, state;
        unsigned        seq, m_seq, r_seq;
        int             last_type;
        unsigned        depth;
        int             total_link_count;
        struct saved {
                struct path link;
                struct delayed_call done;
                const char *name;
                unsigned seq;
        } *stack, internal[EMBEDDED_LEVELS];
        struct filename *name;
        struct nameidata *saved;
        unsigned        root_seq;
        int             dfd;
        kuid_t          dir_uid;
        umode_t         dir_mode;
} __randomize_layout;
```

Схемы алгоритмов

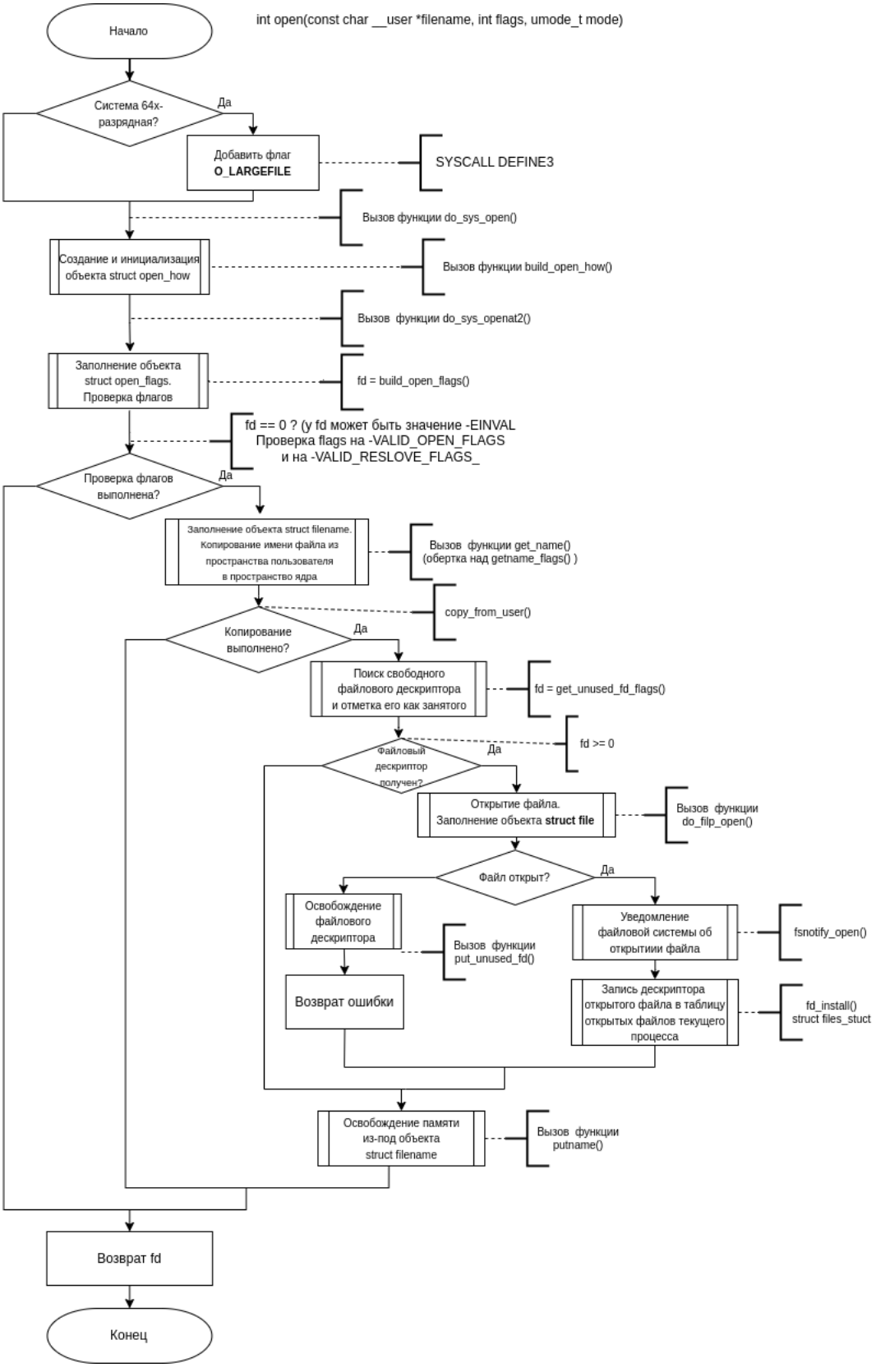

Рисунок 1 – Схема алгоритма работы системного вызова open()

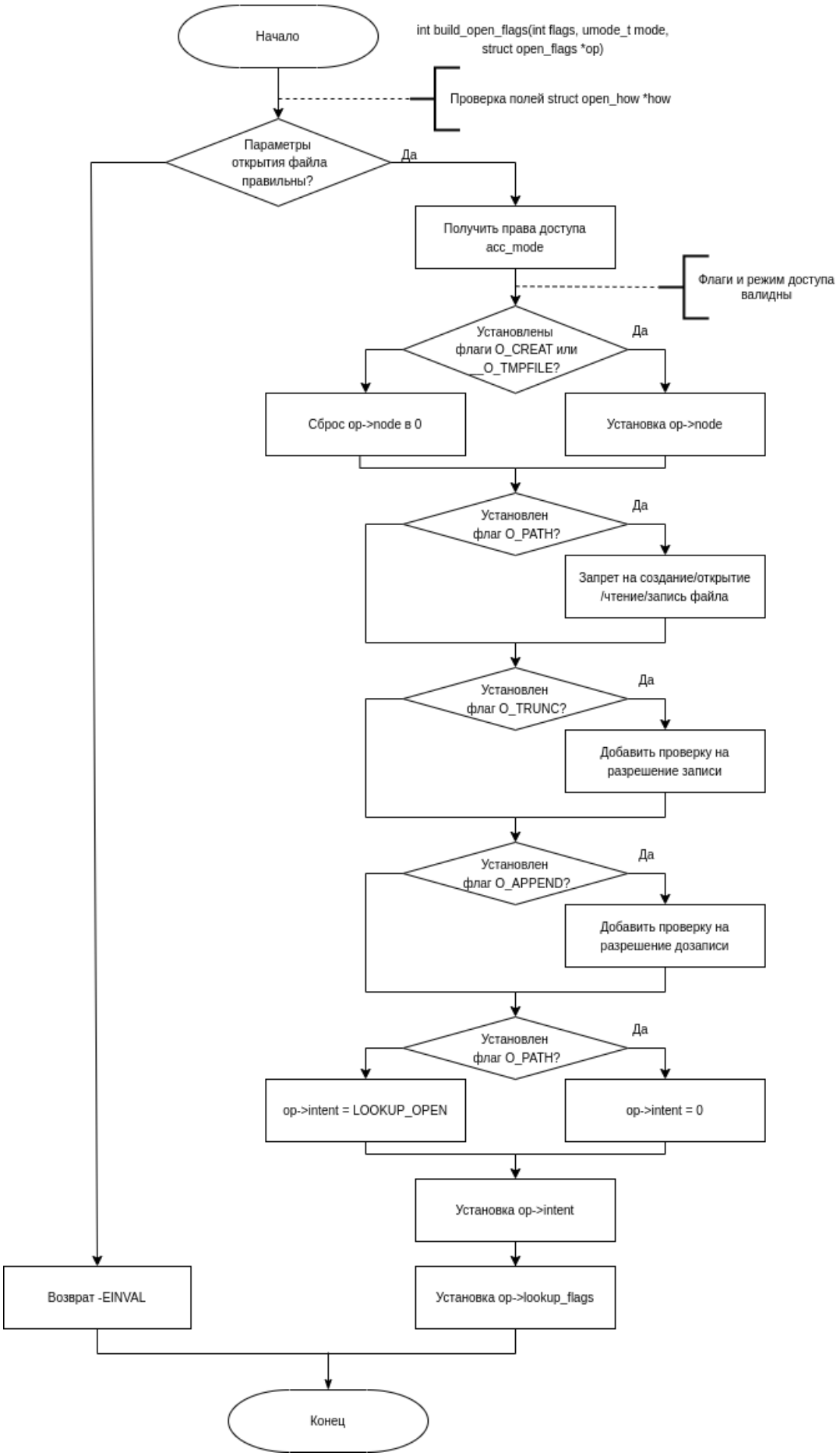

Рисунок 2 – Схема алгоритма работы функции build_open_flags()

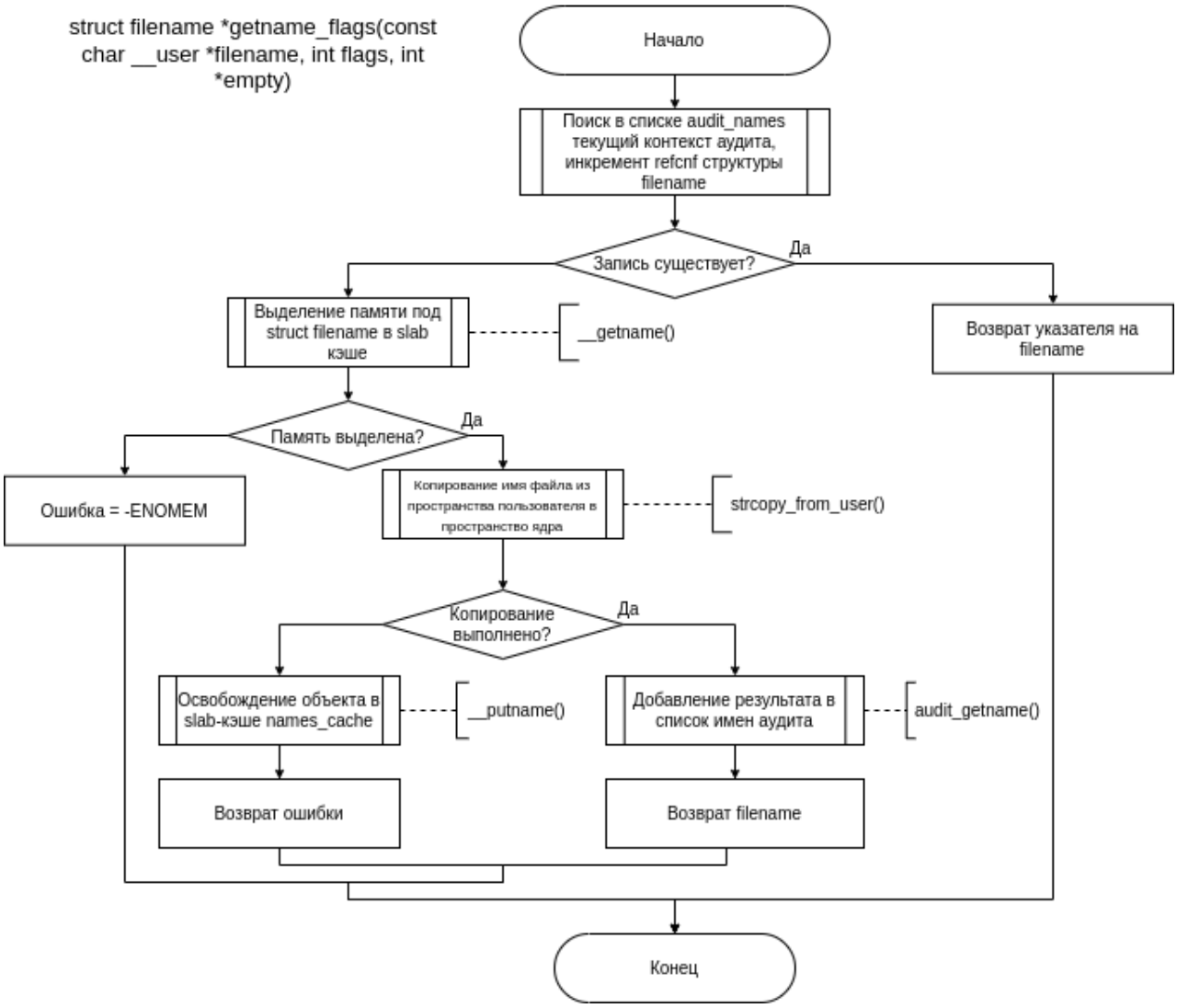

Рисунок 3 – Схема алгоритма работы функции getname_flags()

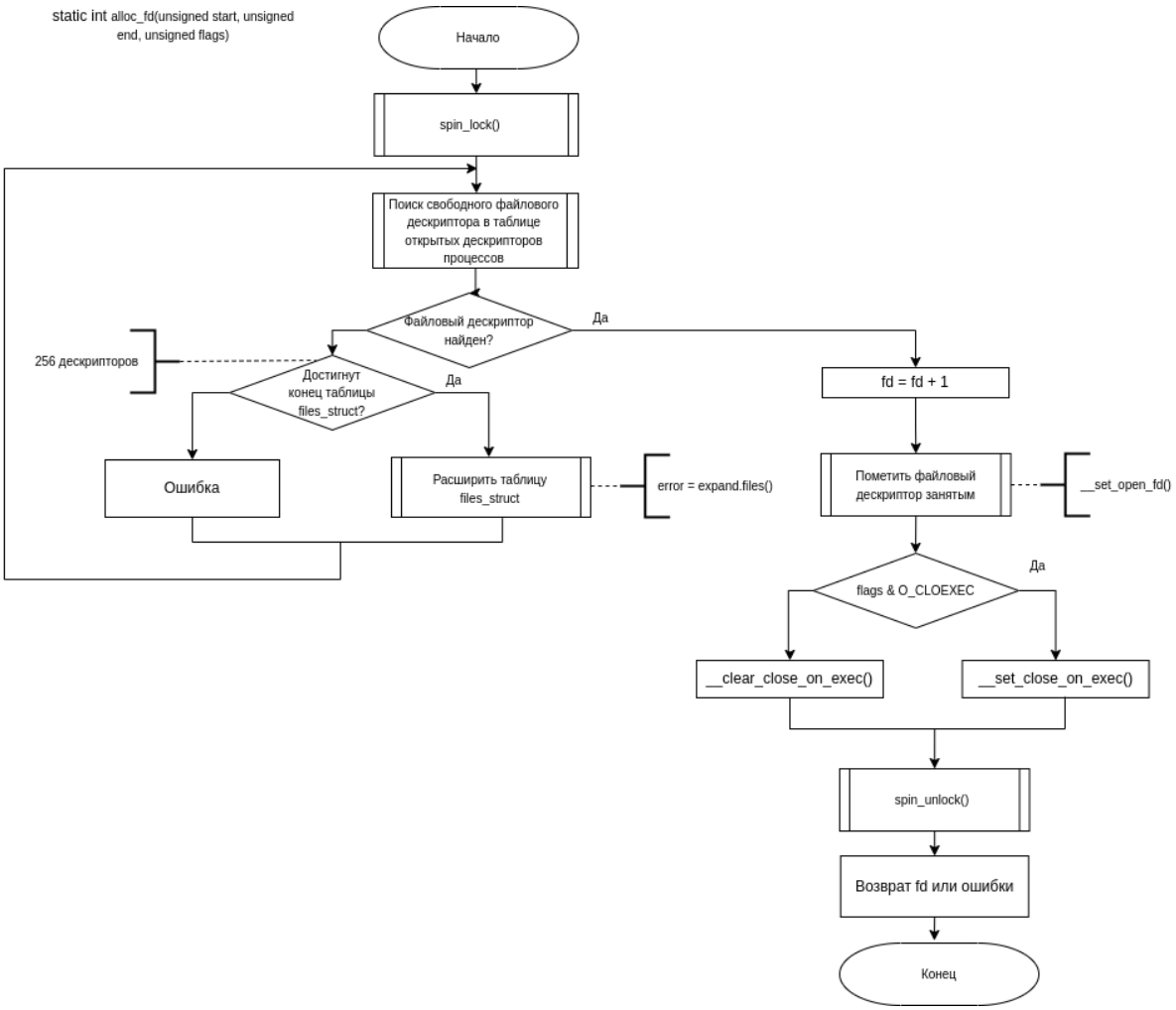

Рисунок 4 – Схема алгоритма работы функции alloc_fd()

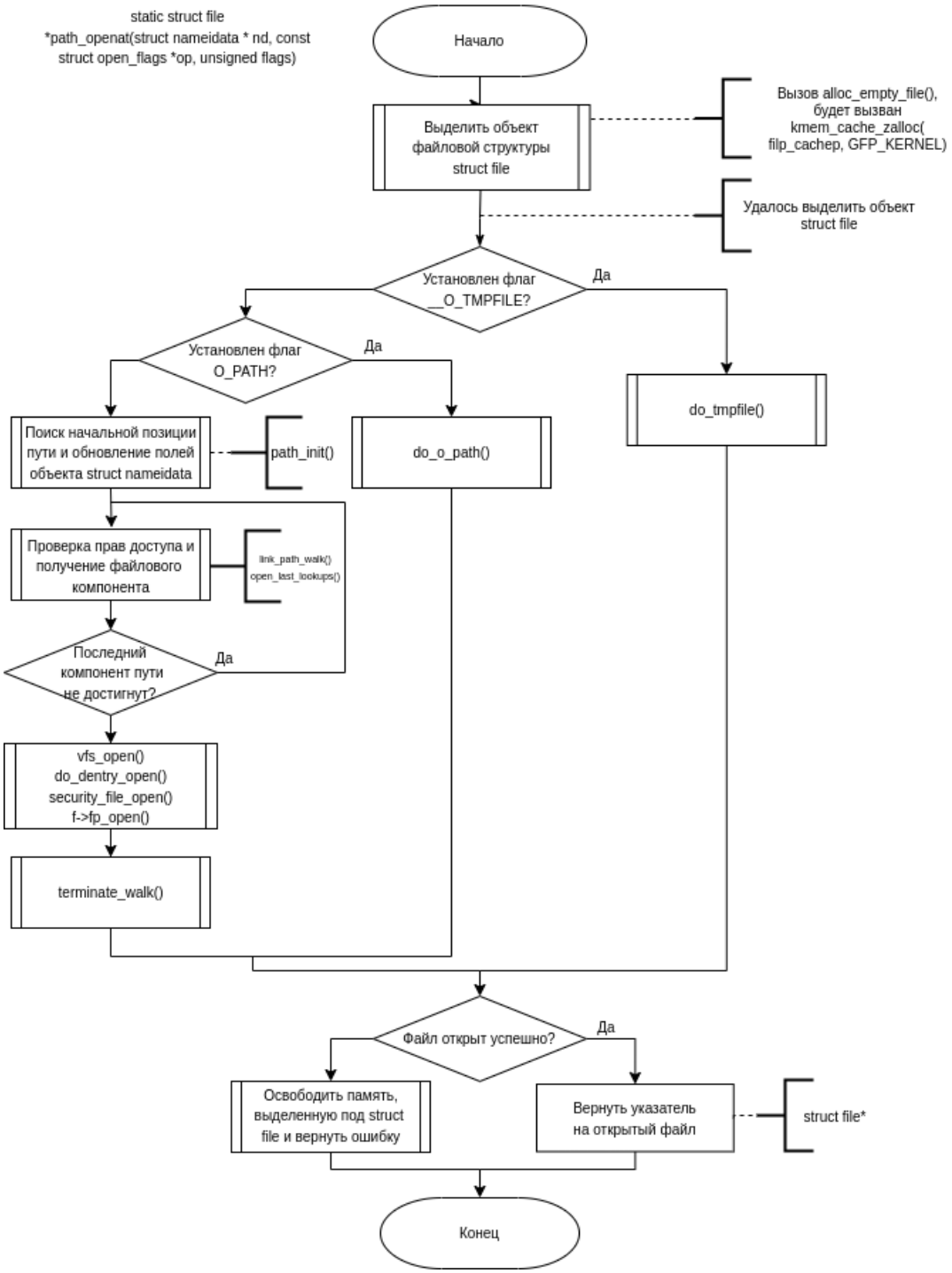

Рисунок 5 – Схема алгоритма функции path_openat()

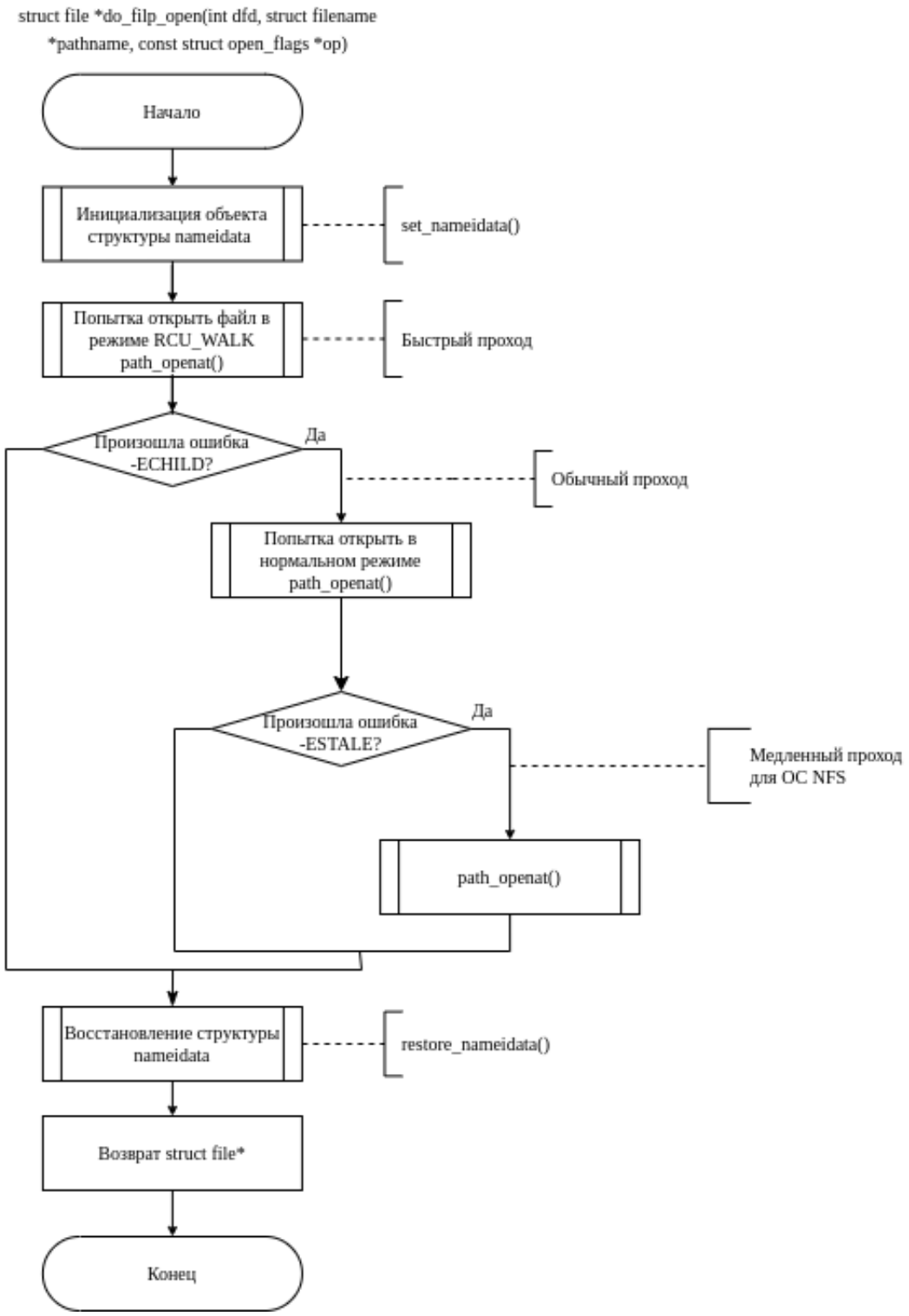

Рисунок 6 – Схема алгоритмов функций, работающих с nameidata (do filp open)

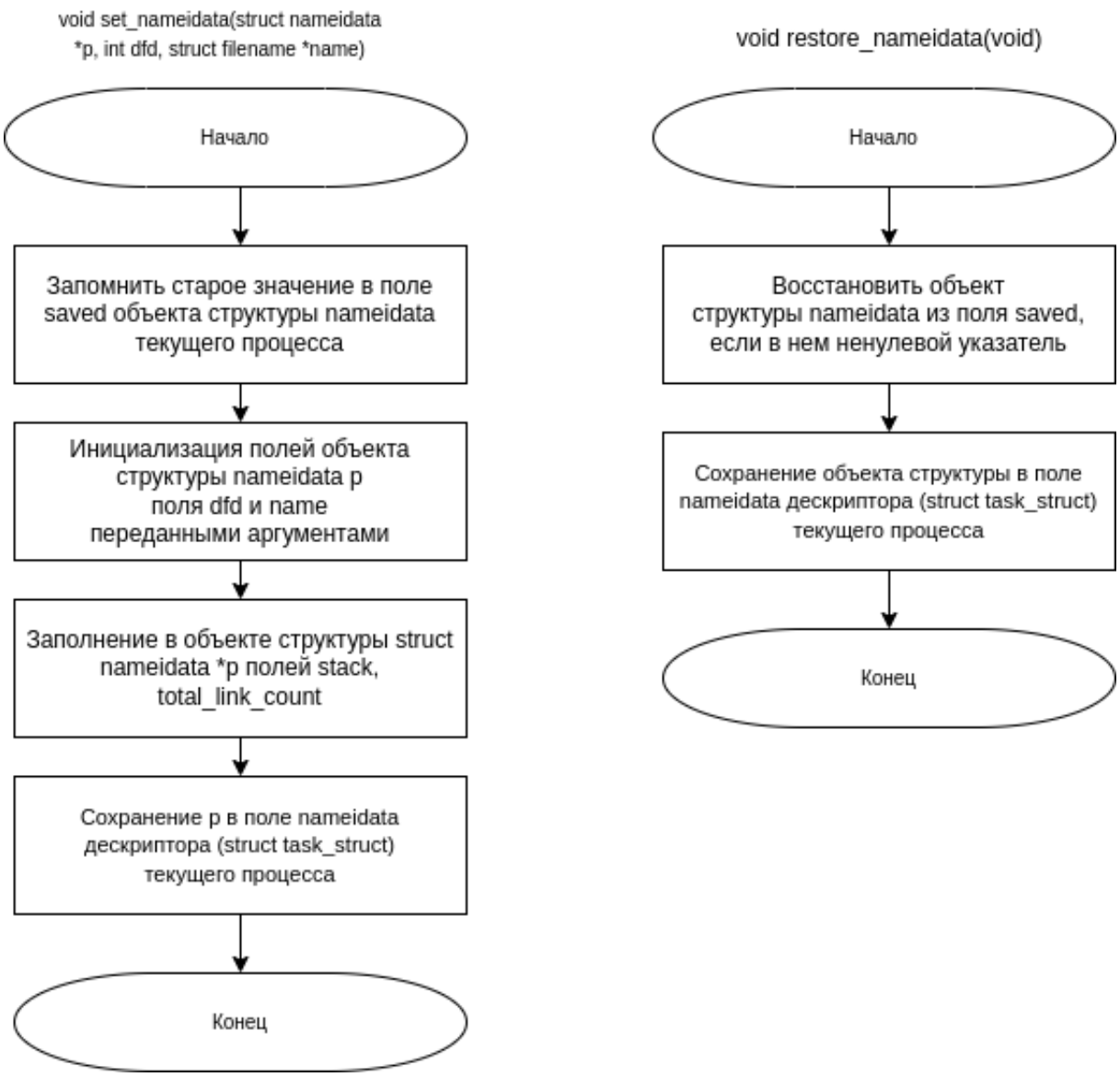

Рисунок 7 – Схема алгоритмов функций, работающих с nameidata

LOOKUP_RCU — флаг для открытия файла в режиме RCU_walk (Допускает возможность одновременного доступа).

LOOKUP_REVAL — флаг для ФС NFS O_APPEND.

O_APPEND может проводить к потери данных файлов в ФС NFS, если одновременно добавл. данные нескольких процессов. Нельзя избежать ускорение гонки.

NFS не поддерживает добавление в файл, потому клиентское ядро имитирует такое поведение.

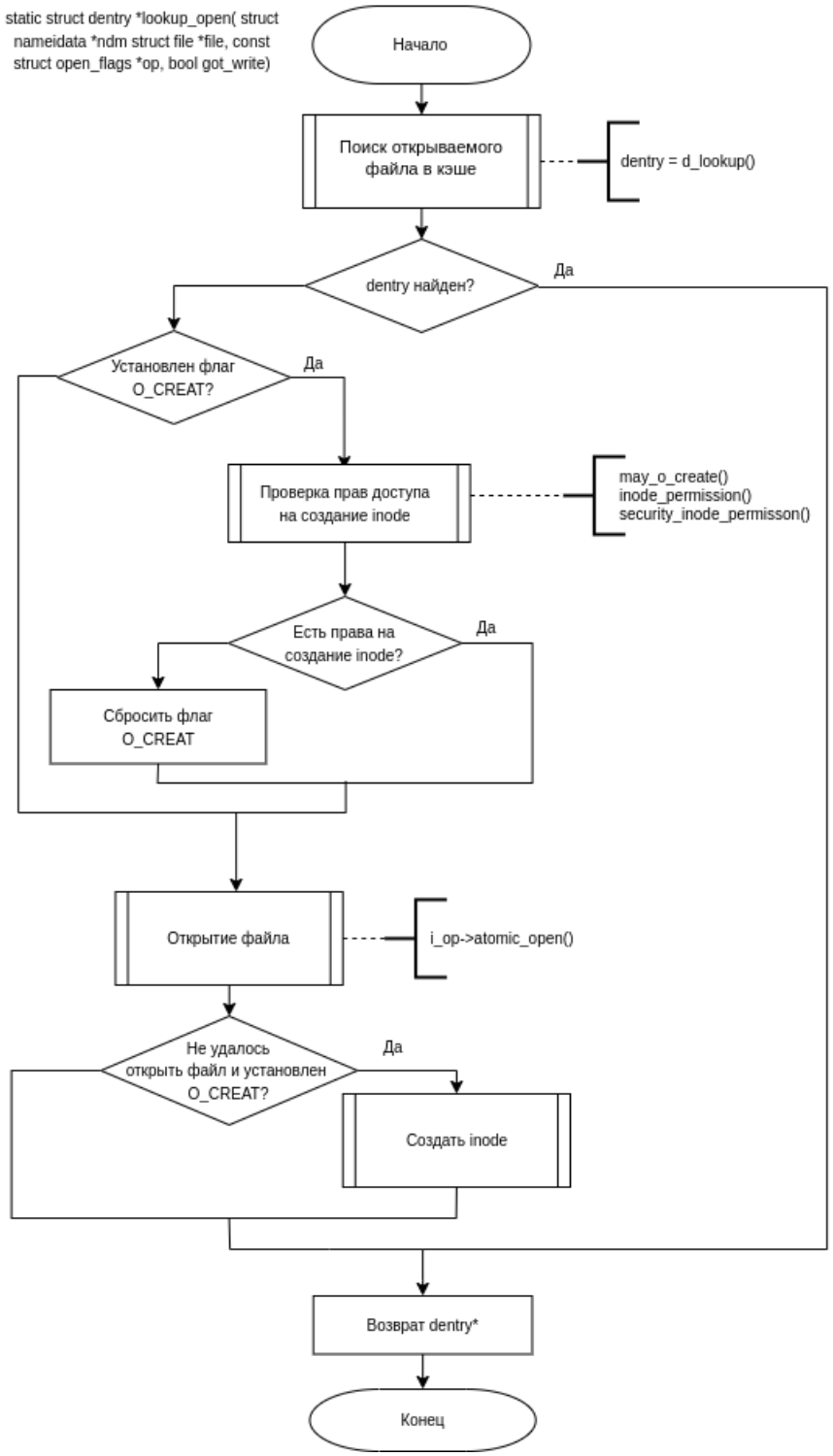

Рисунок 8 – Схема алгоритма функции open_last_lookups()

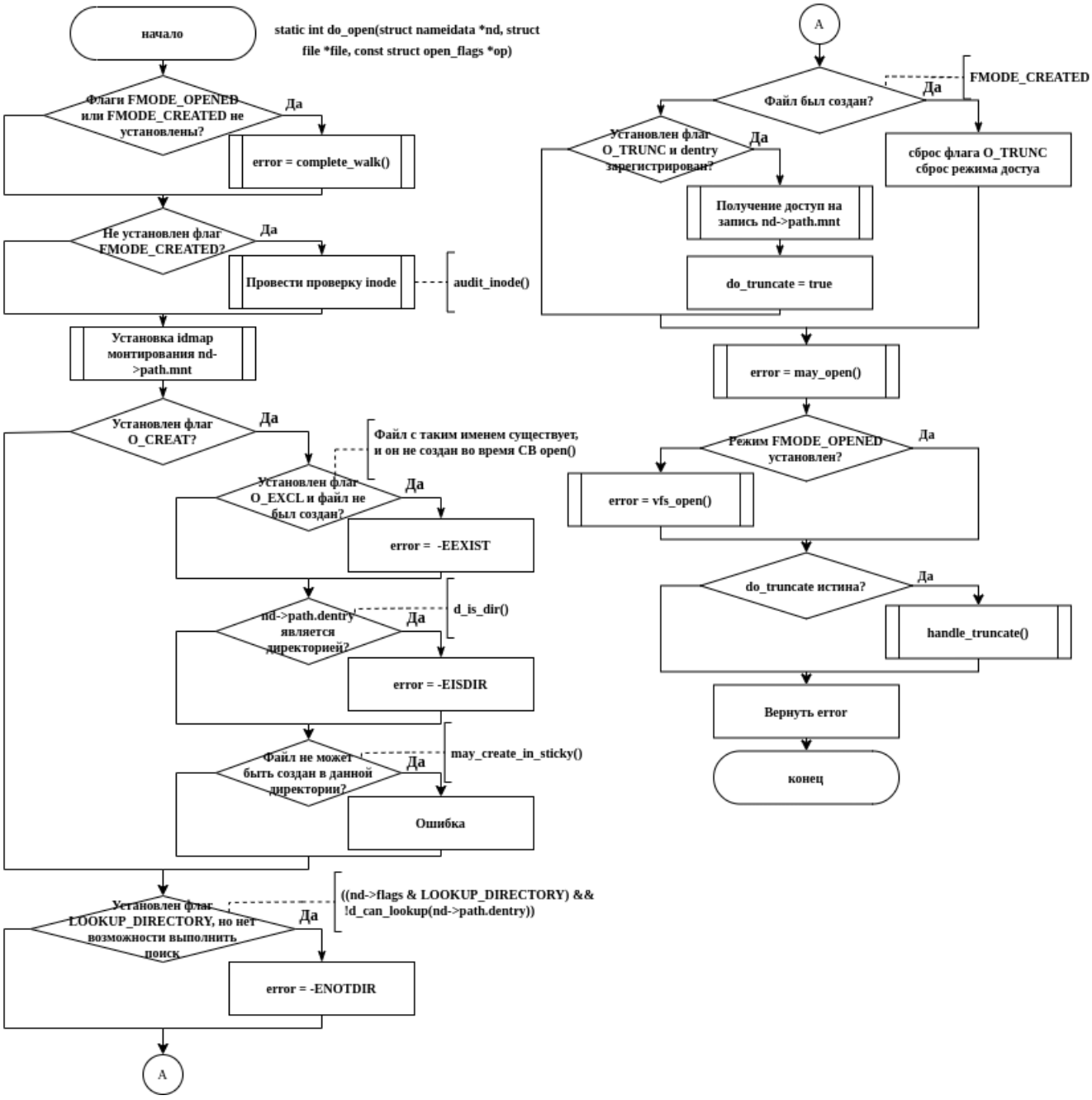

Рисунок 9 – Схема алгоритма функции do_open()

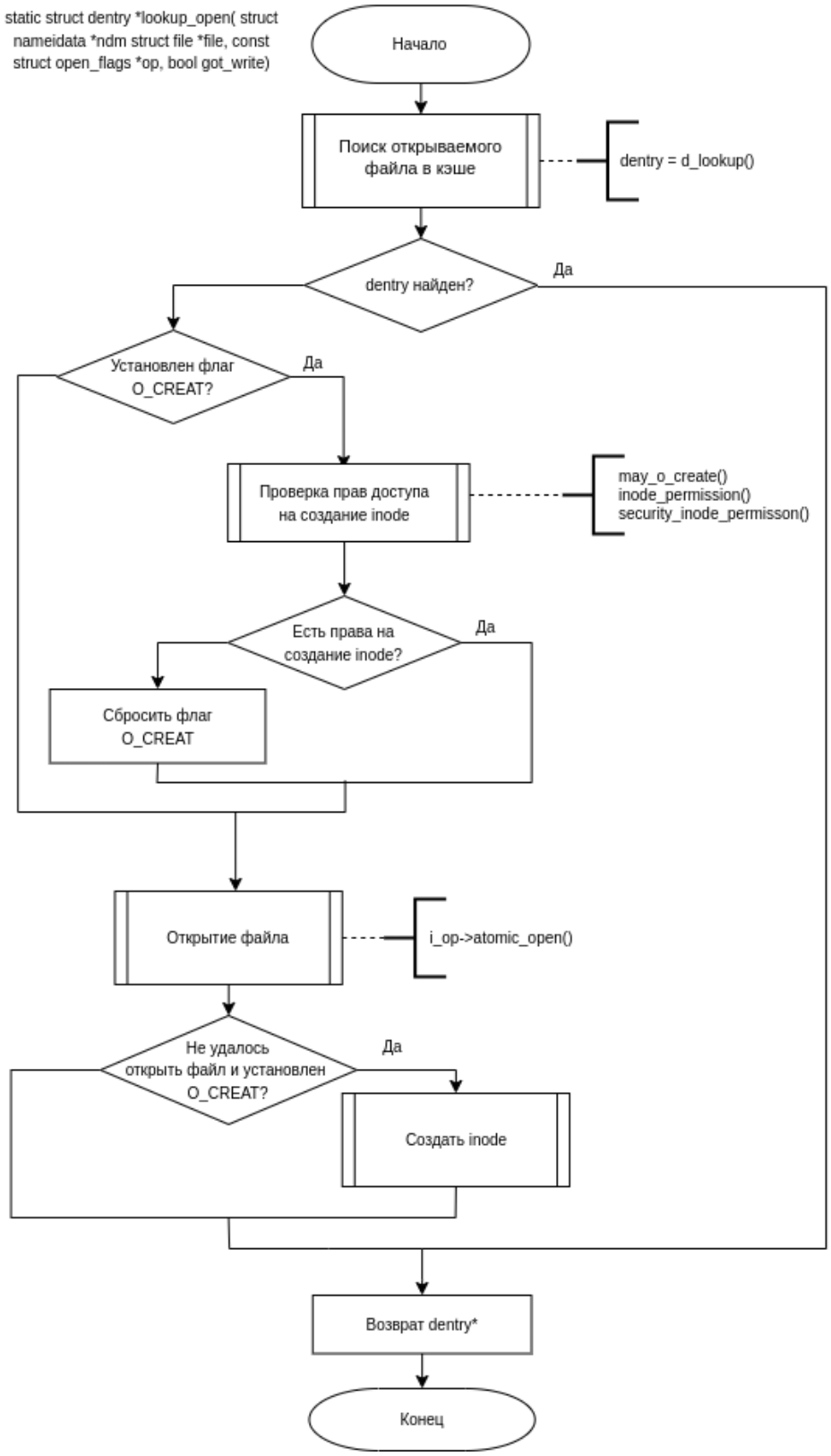

Рисунок 10 – Схема алгоритма функции open_lookup()

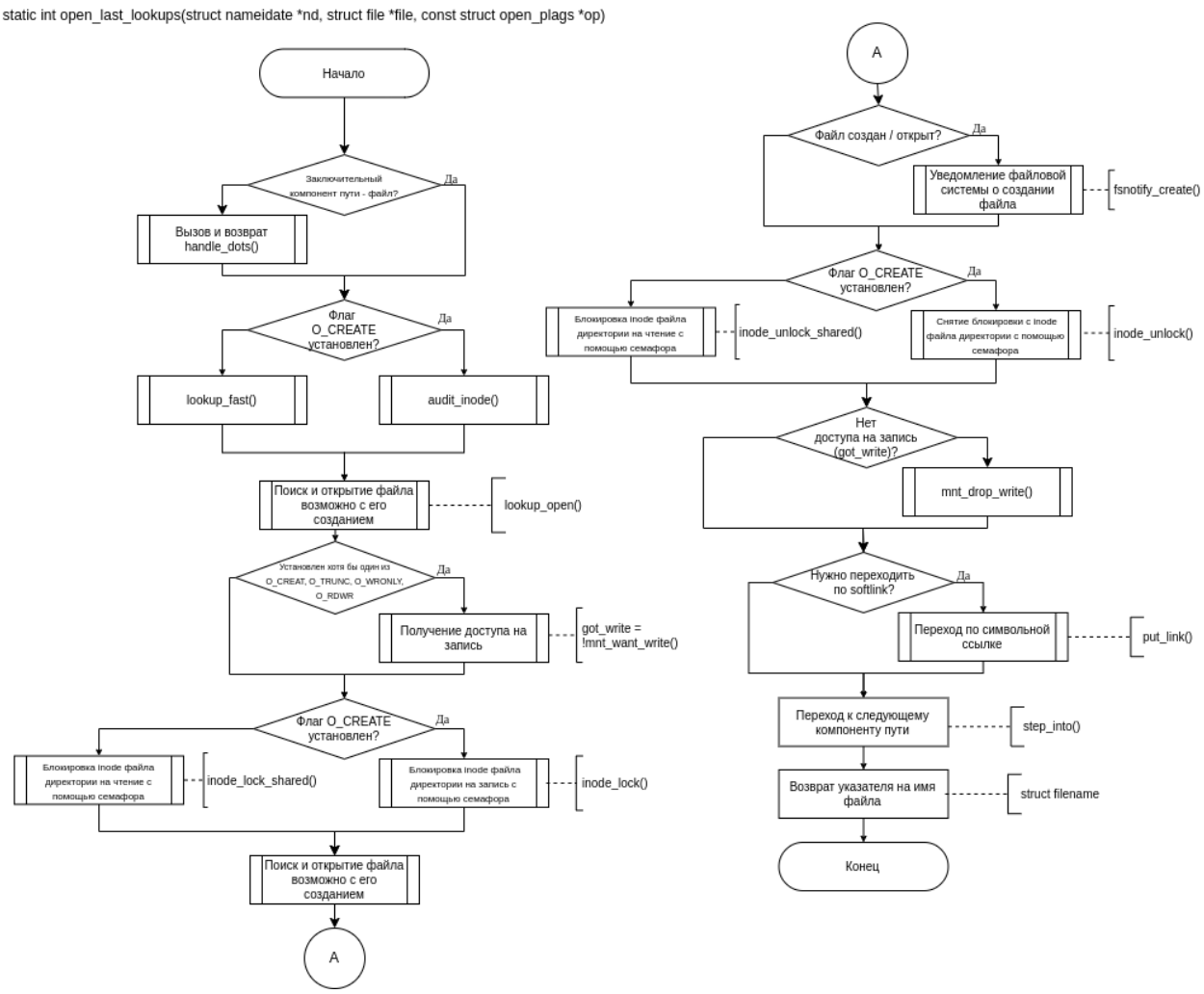

Рисунок 11 – Схема алгоритма функции last_lookup()