

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Кафедра «Компьютерная безопасность»

ОТЧЕТ  
К ЛАБОРАТОРНОЙ РАБОТЕ №1

по дисциплине

«Языки программирования»

Работу выполнила  
студентка группы  
СКБ-232

\_\_\_\_\_  
подпись, дата

Д.В. Иванова

Работу проверил

\_\_\_\_\_  
подпись, дата

С.А. Булгаков

# Содержание

Постановка задачи	3
1 Описание функции main	4
2 Тестирование	4
1 Проверка работы с файлом . . . . .	4
2 Проверка работы с пользовательским вводом . . . . .	4
3 Проверка вывода сообщения об ошибке открытия файла . . . . .	5
Приложение А	6
Приложение Б	8

## Постановка задачи

Необходимо было разработать программу выполняющую шифрование данных по алгоритму Цезаря. Если на вход программе подавалось имя текстового файла, то программа должна была считать текст для шифрования из этого файла, иначе – из пользовательского ввода.

Поскольку не было уточнено, как должны вводиться данные, если работа осуществляется не с файлом, а с текстом, вводимым пользователем, то после запуска программы без названия файла в аргументах, она запрашивает пользователя сообщение, которое нужно закодировать и сдвиг. Сдвиг также запрашивается, если файл был передан в качестве аргумента.

# 1 Описание функции main

## Ввод и работа с аргументами

Если пользователь не вводил имя файла как аргумент при запуске программы, то после запуска программу пользователю предлагается ввести сообщение, которое он хочет закодировать и сдвиг, на который сдвинется алфавит вперёд. Если же пользователь ввёл название файла и такой файл существует, то шифруемый текст считывается из файла, сдвиг пользователь вводит в ответ на запрос программы, после чего создаётся файл enc.txt в который записывается зашифрованное сообщение.

## Алгоритм шифра Цезаря

Алгоритм шифрования Цезаря проверяет вводимый символ на то, заглавная это буква или строчная. После определения, ищется место этой буквы в алфавите, прибавляется сдвиг и проверяется, что не случилось выхода из алфавита, после чего в алфавите ищется буква с новым номером и запоминается как буква зашифрованного слова. Таким образом, шифруется каждая буква вводимого сообщения и получается зашифрованное сообщение.

## 2 Тестирование

### 1 Проверка работы с файлом

Запишем в файл mes.txt "Hello проверим, что файла enc.txt изначально не было. Запустим программу с аргументом mes.txt, введём сдвиг 5. Появился файл enc.txt с зашифрованным по алгоритму шифра Цезаря со сдвигом 5 словом "Hello".

```
$ cat mes.txt
Hello
$ ls
a.out Caesar.c mes.txt README.md
$ ./a.out mes.txt
Input shift:5
$ cat enc.txt
Mjqqt
```

### 2 Проверка работы с пользовательским вводом

После запуска программы пользователя просят ввести сдвиг и сообщение, после чего выводится "Encrypted message:" и сообщение, зашифрованное сообщение с ведённым сдвигом по алгоритму шифрования Цезаря.

```
$ ./a.out
Input shift:5
Input message:Hello
Encrypted message:Mjqqt
```

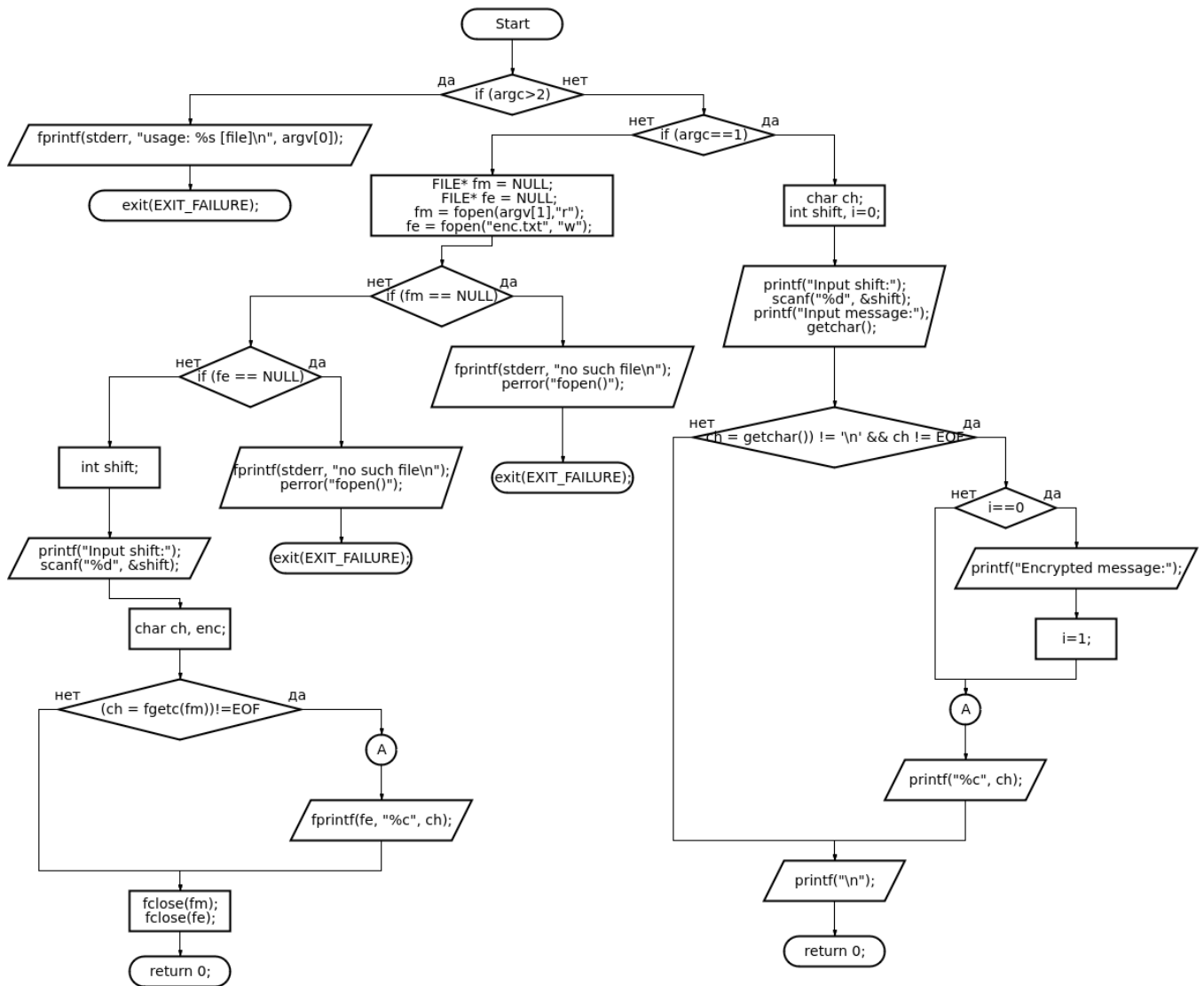
### 3 Проверка вывода сообщения об ошибке открытия файла

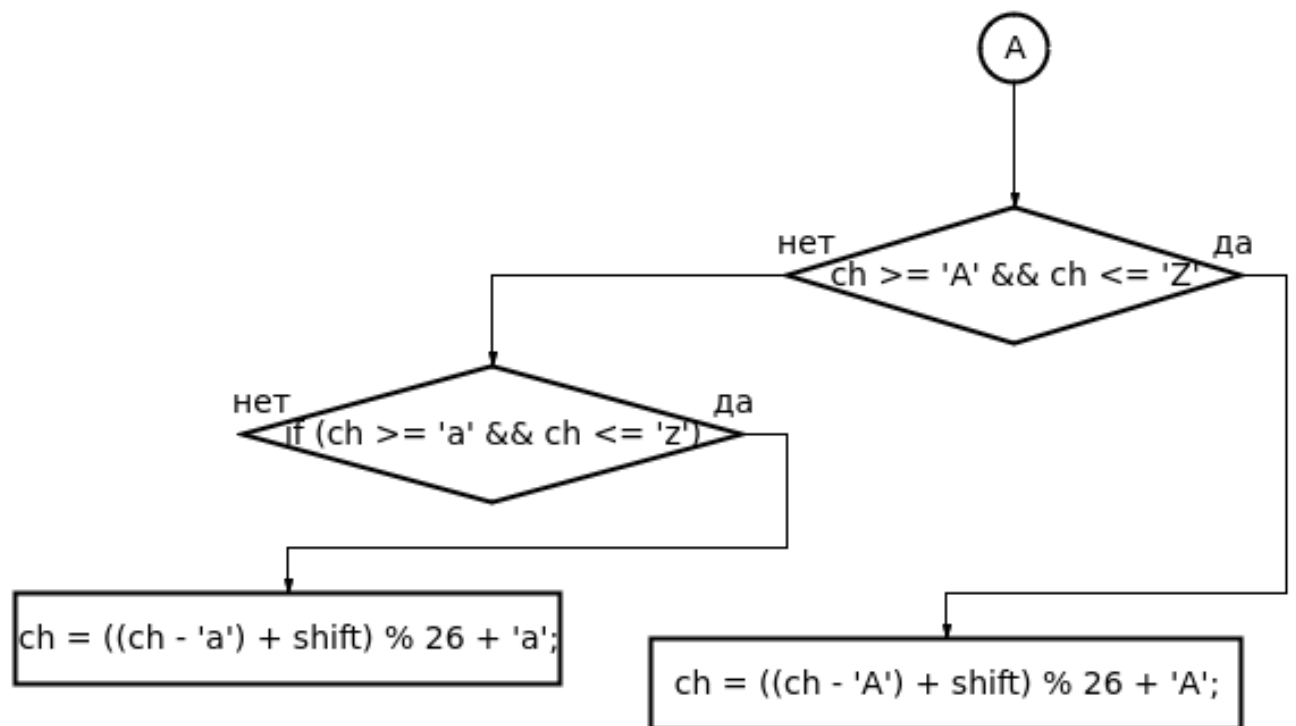
Если файла, который пользователь указал в аргументе к программе нет, то программа выводит ошибку с соответствующим предупреждением.

```
$ ./a.out what.txt
no such file
fopen(): No such file or directory
```

# Приложение А

## Блок-схема *main*





# Приложение Б

## Код программы

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>

int main(int argc, char **argv){
    if (argc>2){
        fprintf(stderr, "usage: %s [file]\n", argv[0]);
        exit(EXIT_FAILURE);
    }
    else{
        if (argc==1){
            char ch;
            int shift, i=0;
            printf("Input shift:");
            scanf("%d", &shift);
            printf("Input message:");
            getchar();
            while ((ch = getchar()) != '\n' && ch != EOF){
                if (i==0){
                    printf("Encrypted message:");
                    i=1;
                }
                if (ch >= 'A' && ch <= 'Z') {
                    ch = ((ch - 'A') + shift) % 26 + 'A';
                }
                else if (ch >= 'a' && ch <= 'z') {
                    ch = ((ch - 'a') + shift) % 26 + 'a';
                }
                printf("%c", ch);
            }
            printf("\n");
        }
        else {
            FILE* fm = NULL;
            FILE* fe = NULL;
            fm = fopen(argv[1], "r");
            fe = fopen("enc.txt", "w");
            if (fm == NULL){
                fprintf(stderr, "no such file\n");
                perror("fopen()");
                exit(EXIT_FAILURE);
            }
            if (fe == NULL){
                fprintf(stderr, "no such file\n");
                perror("fopen()");
                exit(EXIT_FAILURE);
            }
        }

        int shift;
        printf("Input shift:");
        scanf("%d", &shift);

        char ch, enc;
        while ((ch = fgetc(fm))!=EOF){
            if (ch >= 'A' && ch <= 'Z') {
                ch = ((ch - 'A') + shift) % 26 + 'A';
            }
            else if (ch >= 'a' && ch <= 'z') {
                ch = ((ch - 'a') + shift) % 26 + 'a';
            }
            fprintf(fe, "%c", ch);
        }
        fclose(fm);
        fclose(fe);
    }
    return 0;
}
```