

Национальный исследовательский ядерный университет «МИФИ»
Институт интеллектуальных кибернетических систем
Кафедра №12 «Компьютерные системы и технологии»



ОТЧЕТ

О выполнении лабораторной работы No1
«Алгоритмизация обработки целых чисел»

Студент: Николаева А.В.
Группа: Б22-534
Преподаватель: Чепик Н.А.

Москва —2022

1. Формулировка индивидуального задания

Вариант №12 Написать программу, которая переставляет цифры числа так, чтобы сделать его минимальным

2. Описание использованных типов данных

При выполнении данной лабораторной работы использовался встроенный тип данных `int`, предназначенный для работы с целыми числами.

3. Описание использованного алгоритма

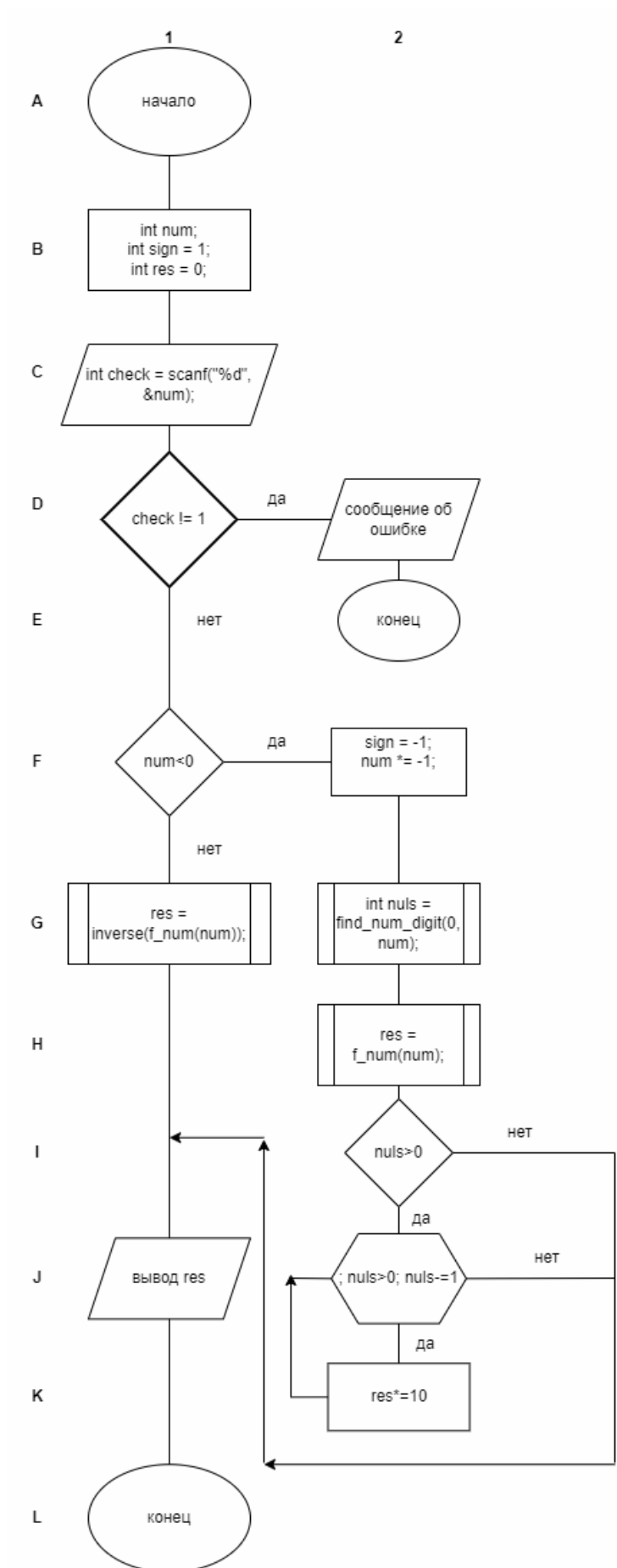


Рис 1: Блок-схема алгоритма работы функции main

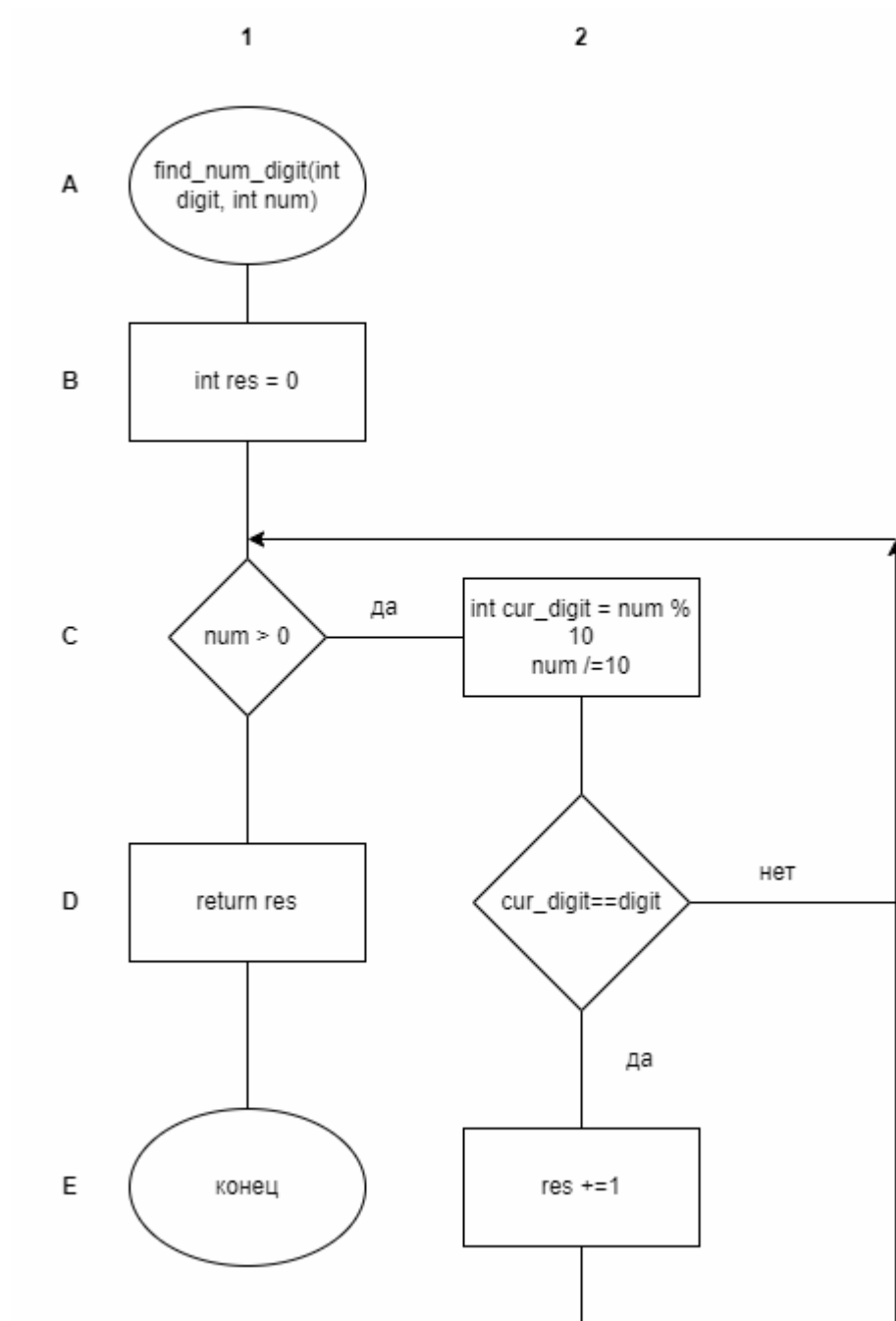


Рис 2: Блок-схема алгоритма работы функции `find_num_digit`

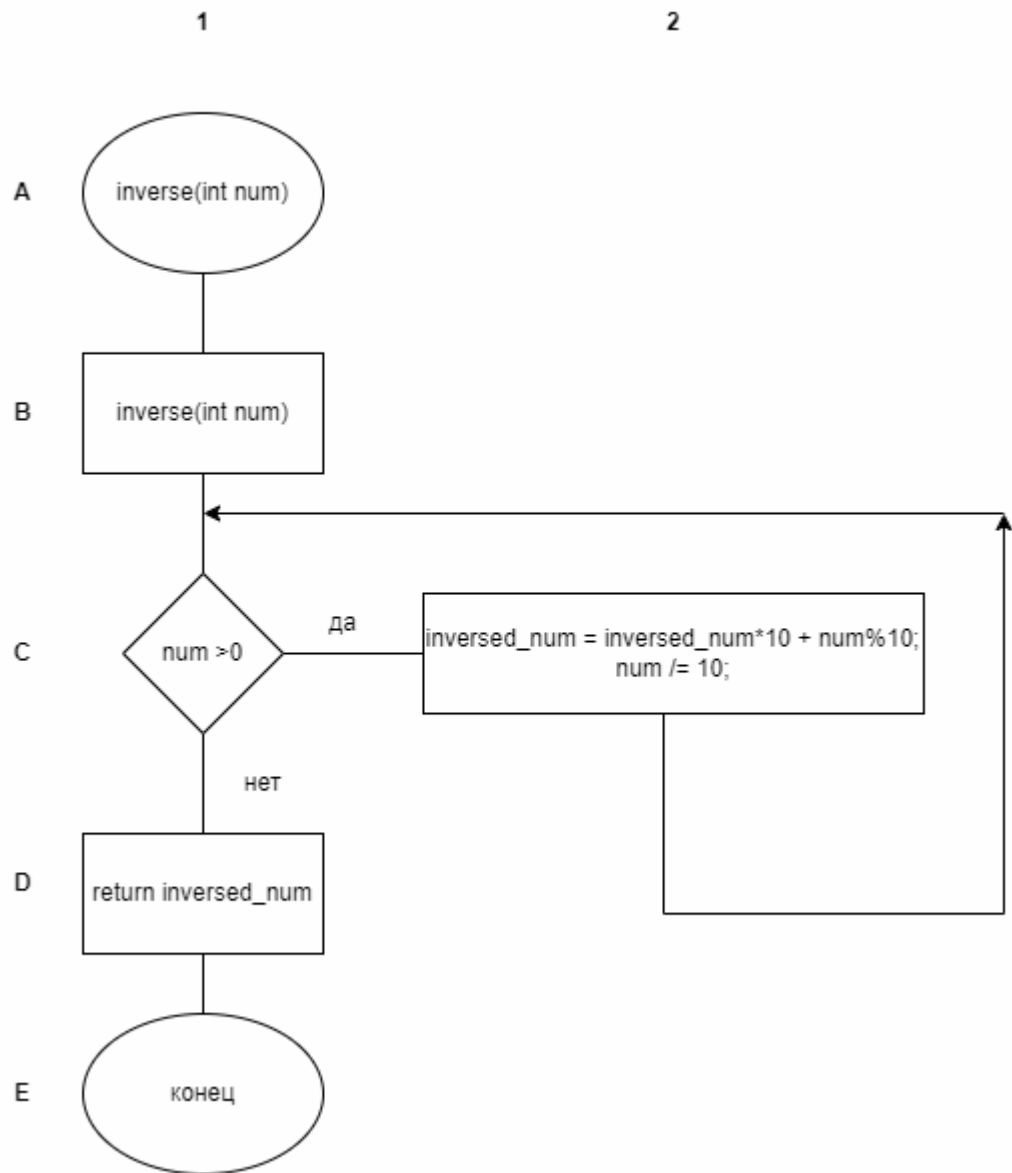


Рис 3: Блок-схема алгоритма работы функции inverse

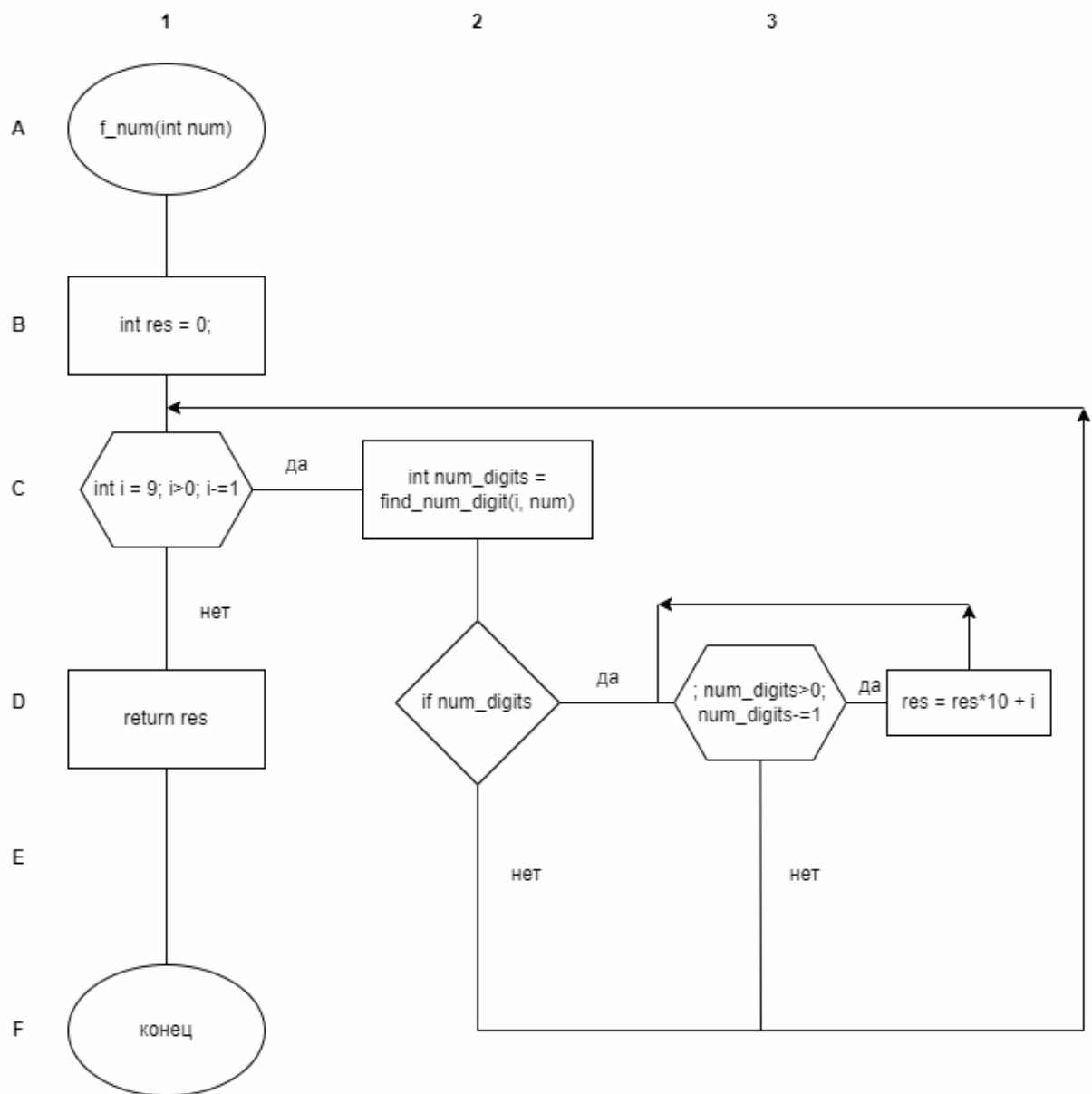


Рис 4: Блок-схема алгоритма работы функции f_num

4. Исходные коды разработанных программ

Листинг 1: Исходные коды программы lab1 (файл: lab1.c)

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

```
int inverse(int num){
int inversed_num = 0;
while (num){
    inversed_num = inversed_num*10 + num%10;
    num /= 10;
}
return inversed_num;
}
```

```

int find_num_digit(int digit, int num){
int res = 0;
while (num){
    int cur_digit = num % 10;
    if (cur_digit==digit){
        res +=1;
    }
    num /= 10;
}
return res;
}

```

```

int f_num(int num){
int res = 0;
for (int i = 9; i>0; i-=1){
    int num_digits = find_num_digit(i, num);
    if (num_digits){
        for (; num_digits>0; num_digits-=1){
            res = res*10 + i;
        }
    }
}
return res;
}

```

```

int main(void){
int num;
int sign = 1;
int res = 0;
int check = scanf("%d", &num);
if (check != 1){
    printf("You can write only integer");
    return 1;
}
if (num<0){
    sign = -1;
    num *= -1;
    int nuls = find_num_digit(0, num);
    res = f_num(num);
    if (nuls){
        for (; nuls>0; nuls-=1){
            res*=10;
        }
    }
}
else{
    res = inverse(f_num(num));
}
printf("%d\n", res*sign);
return 0;
}

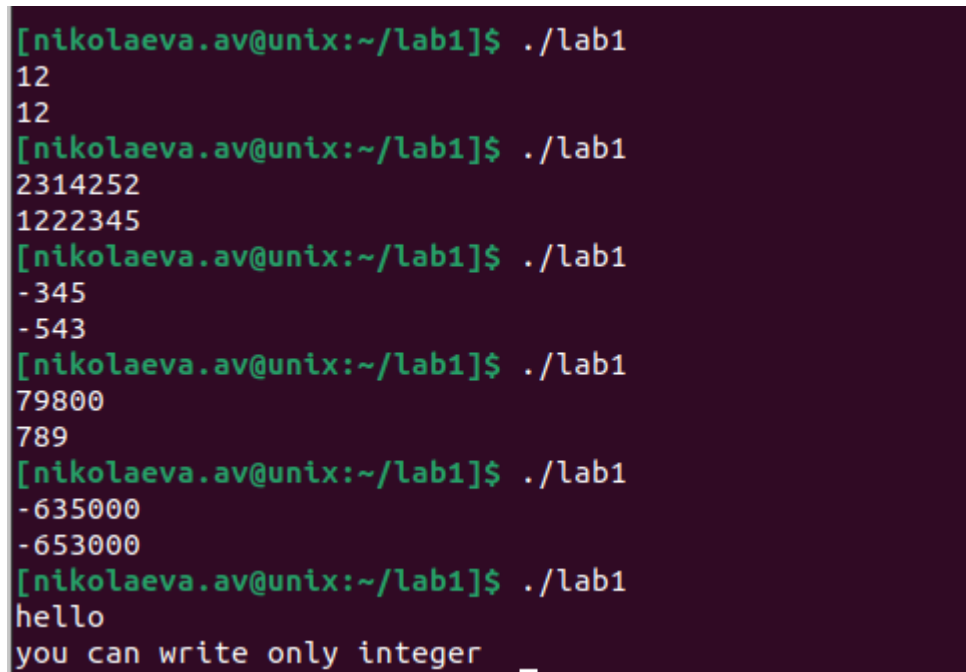
```

5. Описание тестовых примеров

Таблица 1: Тестовые примеры

значение num	ожидаемый результат	полученный результат
12	12	12
2314252	12223345	1222345
-345	-543	-543
79800	789	789
-635000	-653000	-653000
hello	ошибка	ошибка

6. Скриншоты



```
[nikolaeva.av@unix:~/lab1]$ ./lab1
12
12
[nikolaeva.av@unix:~/lab1]$ ./lab1
2314252
1222345
[nikolaeva.av@unix:~/lab1]$ ./lab1
-345
-543
[nikolaeva.av@unix:~/lab1]$ ./lab1
79800
789
[nikolaeva.av@unix:~/lab1]$ ./lab1
-635000
-653000
[nikolaeva.av@unix:~/lab1]$ ./lab1
hello
you can write only integer
```

Рис 5: результаты тестов

7. Выводы

В ходе выполнения данной работы на примере программы, выполняющей перестановку цифр числа, были рассмотрены базовые принципы работы построения программ на языке C и обработки целых чисел:

1. Организация ввода/вывода.
2. Разработка функций.
3. Объявление и использование переменных.
4. Выполнение простейших арифметических операций над целочисленными операндами.