

Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №12 «Компьютерные системы и технологии»



# ОТЧЕТ

## О выполнении лабораторной работы №1 «Алгоритмизация обработки целых чисел»

Студент: Гаврилов Д.А.

Группа: Б23-516

Преподаватель: Бабалова И.Ф.

Москва – 2023

# 1. Формулировка индивидуального задания

Вариант №7. Дано целое число. Поменять местами цифры числа так, чтобы получить минимальное число из этих цифр.

## 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

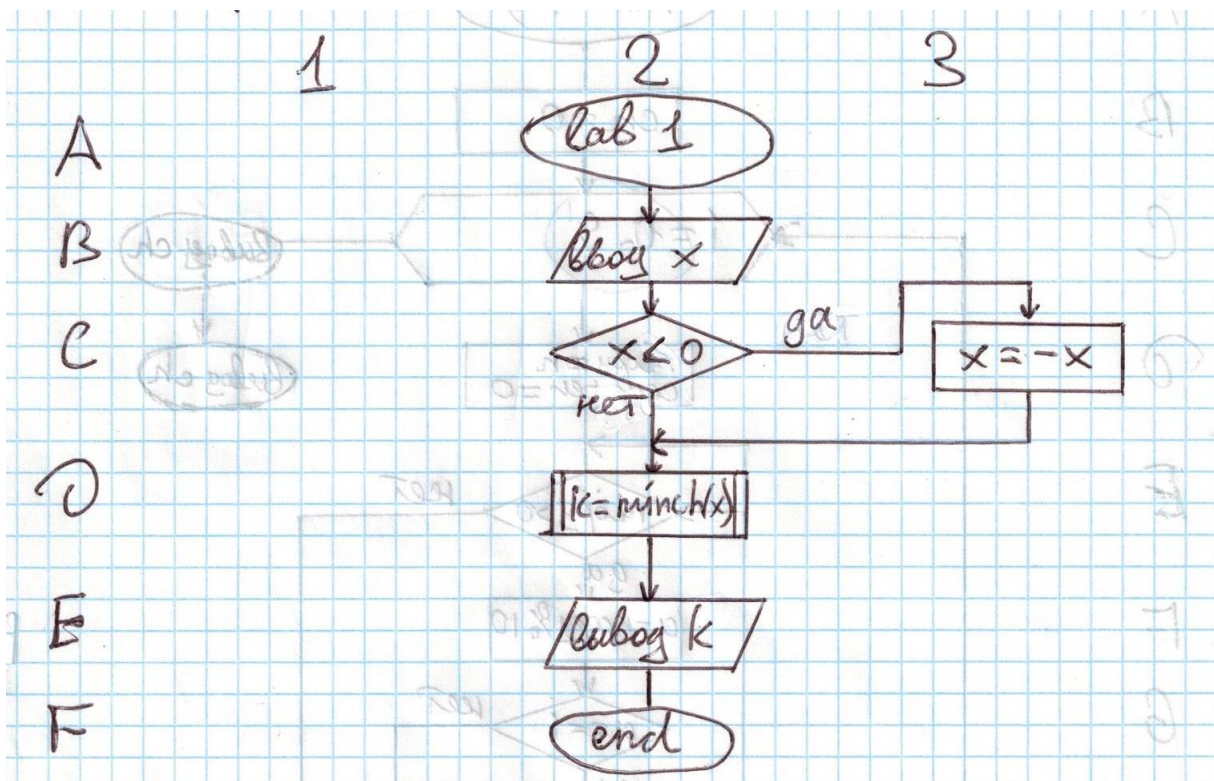


Рис. 1: Блок-схема алгоритма работы функции `main()`

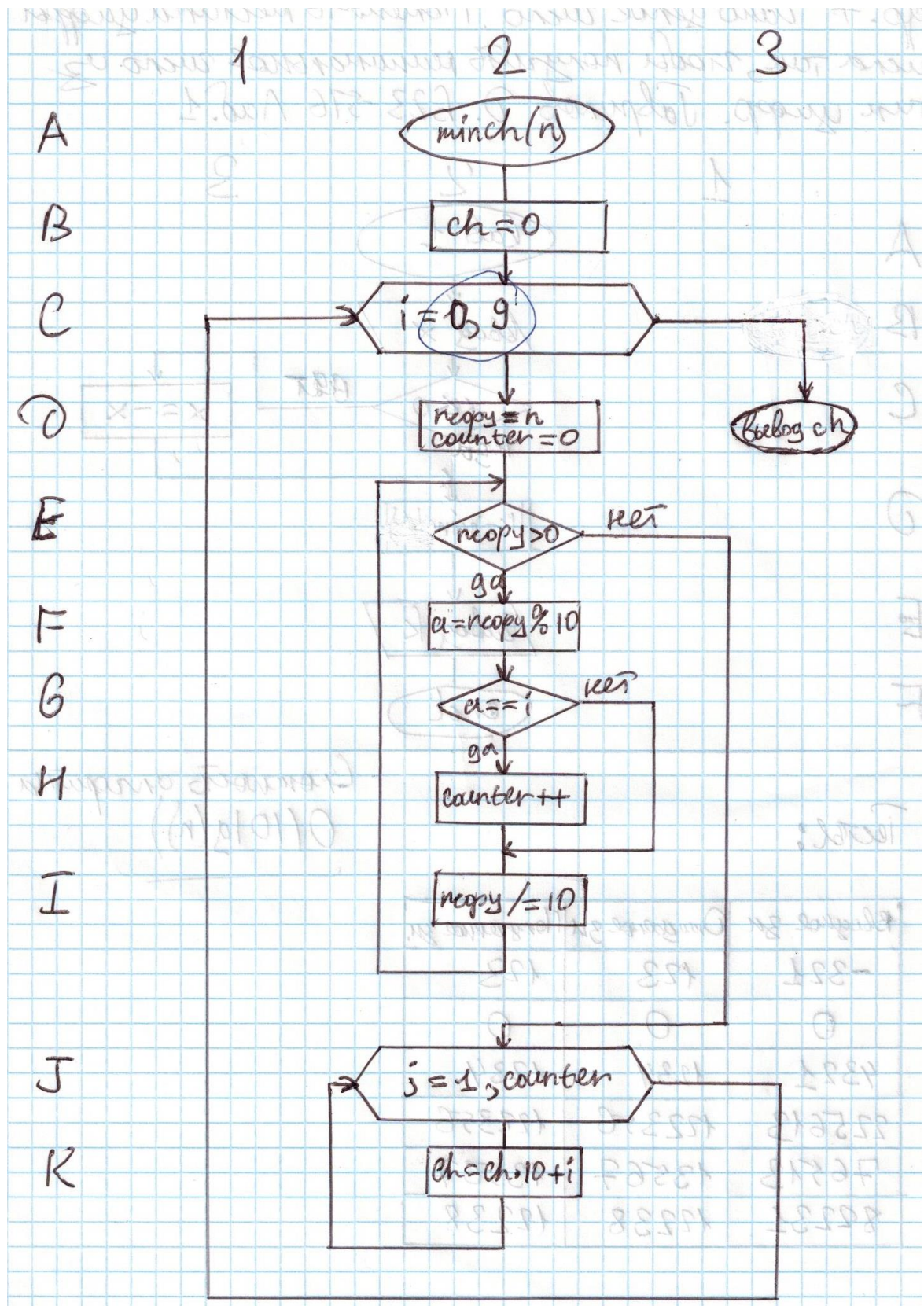


Рис. 2: Блок-схема алгоритма работы функции minch()

## 4. Исходные коды разработанных программ

Листинг 1: Исходные коды программы reslab1(файл: lab1.c)

```
#include <stdio.h>

int minch(int n)
{
    int ch = 0;
    for(int i = 0; i <= 9; i++) // перебор цифр от 0 до 9, считаем количество
                               // цифры i в числе
    {
        int ncopy = n; // создаем копию числа n, используем её для подсчёта цифры i
        int counter = 0;
        while (ncopy > 0) // цикл подсчёта цифры i в ncopy
        {
            int a = ncopy % 10;
            if (a == i) {counter++;}
            ncopy /= 10;
        }
        for(int j = 1; j <= counter; j++) {
            ch = ch * 10 + i; // прибавляем к результату ch цифру i counter раз
        }
    }
    return ch; // вернуть результат
}

int main()
{
    int x,k;
    printf("Введите x\n");
    scanf("%d", &x); // ввод числа на вход
    if (x < 0)
    {
        x = -x; // поменять знак числа, если оно меньше 0
    }
    k = minch(x); // записываем в переменную k результат работы функции minch(x)
    printf("Полученное значение: ");
    printf("%d\n",k); // вывод результата
    return 0;
}
```

## 5. Описание тестовых примеров

Таблица 1: Тестовые примеры

Значение x	Ожидаемое значение k	Полученное значение k
-321	123	123
4321	1234	1234
0	0	0
225613	122356	122356
-654021	12456	12456
822310	12238	12238

## 6. Скриншоты

```
[gavrilov.da@unix lab1]$ cc -o reslab1 lab1.c
[gavrilov.da@unix lab1]$ ./reslab1
Введите x
-321
Полученное значение: 123
[gavrilov.da@unix lab1]$ ./reslab1
Введите x
4321
Полученное значение: 1234
[gavrilov.da@unix lab1]$ ./reslab1
Введите x
0
Полученное значение: 0
[gavrilov.da@unix lab1]$ ./reslab1
Введите x
225613
Полученное значение: 122356
[gavrilov.da@unix lab1]$ ./reslab1
Введите x
-654021
Полученное значение: 12456
[gavrilov.da@unix lab1]$ ./reslab1
Введите x
822310
Полученное значение: 12238
[gavrilov.da@unix lab1]$ |
```

Рис. 3: Сборка и запуск программы reslab

## 7. Выводы

В ходе выполнения данной работы на примере программы, составляющей минимальное число из цифр заданного числа, были рассмотрены базовые принципы работы построения программ на языке C и обработки целых чисел, а также условные операторы, циклы с предусловием и циклы с заданным количеством повторений:

1. Организация ввода/вывода.
2. Разработка функций.

3. Объявление и использование переменных.
4. Выполнение арифметических операций над целочисленными операндами.
5. Использование условного оператора if-else.
6. Использование циклов while и for.