

Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №12 «Компьютерные системы и технологии»



ОТЧЁТ

О выполнении лабораторной работы № 3

"Работа с массивами данных."

Студент: Ким В.А.

Группа: Б22-703

Преподаватель: Овчаренко Е.С.

Москва - 2022

1. Формулировка индивидуального задания

Вариант № 13.

Вариант №13

Индивидуальное задание

В исходной последовательности целых чисел найти те, которые являются простыми. Сформировать из данных чисел новую последовательность, удалив их из исходной.

2. Описание использованных типов данных

В данной работе использовались тип данных «int» для работы с целыми числами. Структура Array

```
struct Array {  
    int* ptr;  
    int count;  
    int size;  
};
```

3. Описание использованного алгоритма

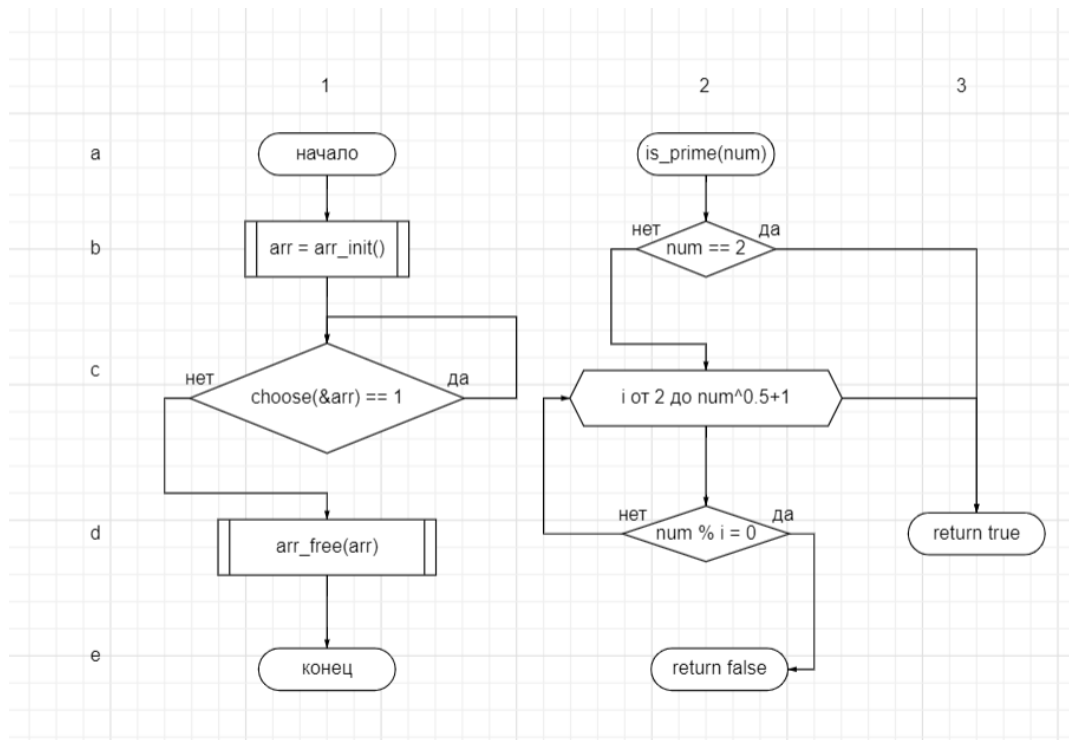


Рис. 1: Блок-схема алгоритма работы функции main и is_prime

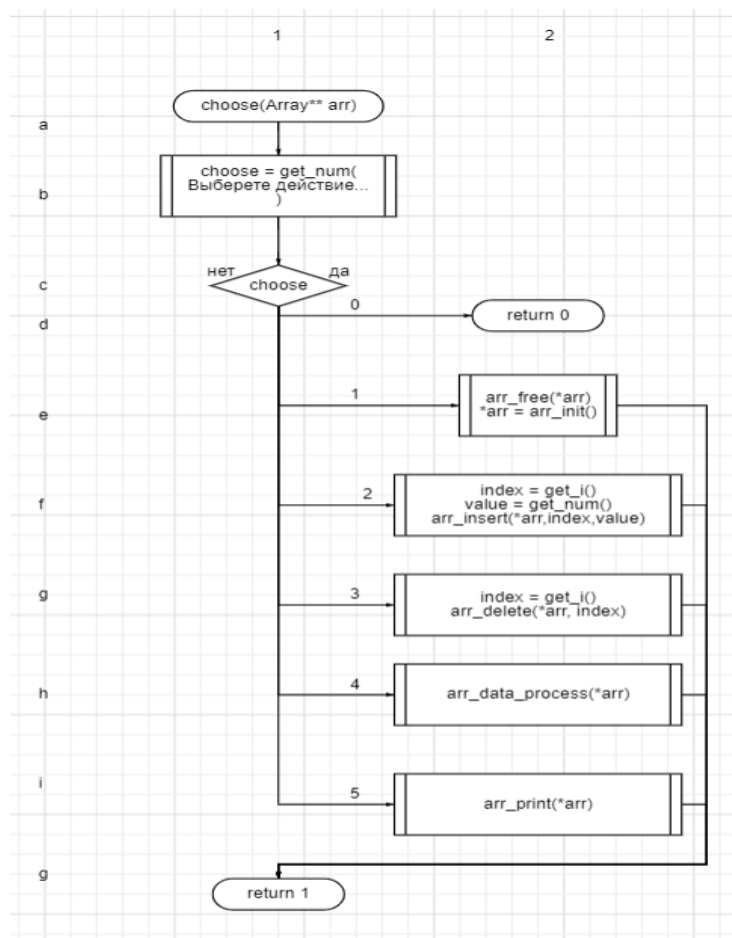


Рис. 2: Блок-схема алгоритма работы функции choose

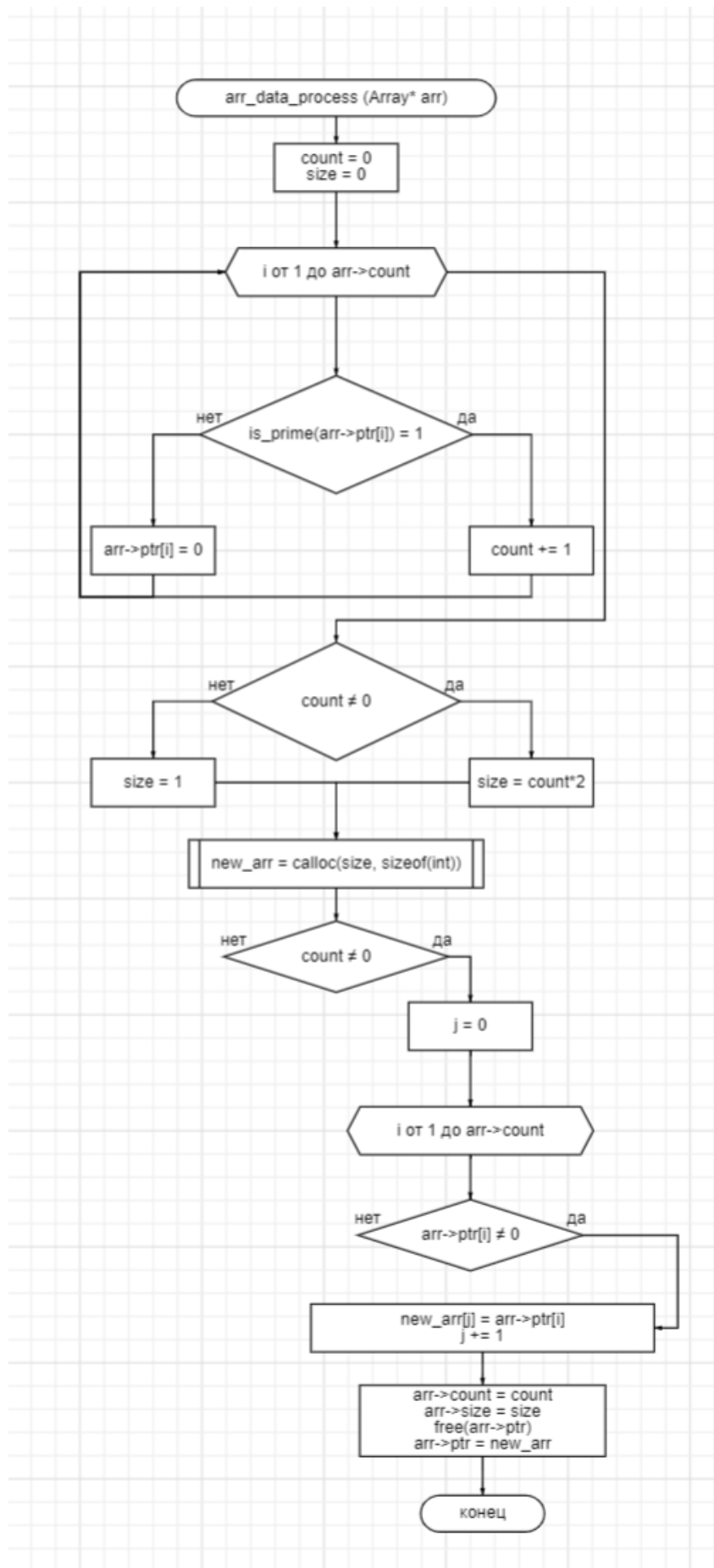


Рис. 3: Блок-схема алгоритма работы функции arr_data_process

4. Исходные коды разработанных программ

Листинг 1: Исходные код программы main, файл: lab_03.c

```
1  #include <stdbool.h>
2  #include <math.h>
3  #include <locale.h>
4
5  #include "lib_arr.h"
6  #include "lib_scanf.h"
7
8  bool is_prime(int num);
9
10 void arr_data_process(Array* arr);
11 int choose(Array** arr);
12
13 int main() {
14     setlocale(LC_ALL, "rus");
15
16     int count = get_i("Введите размер массива: ");
17     Array* arr = arr_init(count);
18     printf("\n\n");
19     arr_print(arr);
20
21     for (;choose(&arr);) {};
22     arr_free(arr);
23
24     return 0;
25 }
26 bool is_prime(int num) {
27     bool flag = true;
28     for (int i = 2; i < sqrt(num)+1; ++i) {
29         if ( num % i == 0 ) {
30             flag = false;
31             break;
32         }
33     }
34
35     return flag;
36 }
37 void arr_data_process(Array* arr) {
```

```

38  int count = 0;
39  for (int i = 0; i < arr->count; ++i) {
40      if (is_prime(arr->ptr[i])) ++count;
41      else arr->ptr[i] = 0;
42  }
43  int size = count ? count * 2 : 1;
44
45  int* new_arr = (int*)calloc(size, sizeof(int));
46  if (!new_arr) exit(1); // throw
47
48  if (count) for (int i = 0, j = 0; i < arr->count; ++i)
49  if (arr->ptr[i]) new_arr[j++] = arr->ptr[i];
50
51  arr->count = count;
52  arr->size = size;
53  free(arr->ptr);
54  arr->ptr = new_arr;
55 }
56
57 int choose(Array** arr) {
58     int choose = get_num("\n\n\
59 Выберите действие: \n\
60 0: Завершить программу\n\
61 1: Пересоздать массив\n\
62 2: Вставить новый элемент в массив\n\
63 3: Удалить элемент\n\
64 4: Обработка данных\n\
65 5: Вывести массив\n\
66 \
67 >>> ");
68     printf("\n\n");
69
70     switch (choose) {
71         case EXIT: return 0;
72
73         case REINIT:
74             arr_free(*arr);
75             int count = get_i("Введите размер массива: ");
76             *arr = arr_init(count);
77             return 1;
78
79         case INSERT: {
80             int index = get_i("Индекс вставляемого элемента: ");
81             int value = get_num("Значение элемента: ");
82             arr_insert(*arr, index, value);
83             return 1;
84         }
85
86         case DELETE: {
87             int index = get_i("Индекс удаляемого элемента: ");
88             arr_delete(*arr, index);

```

```

89     return 1;
90 }

91 case DATA_PROCESS:
92     arr_data_process(*arr);
93     return 1;
94
95 case PRINT:
96     arr_print(*arr);
97     return 1;
98
99 default: return 1;
100 }
101 }

```

5. Описание текстовых примеров

Программа №1

Значения массива	Ожидаемые значения	Полученное значение В
1,2,3,4,5	1,2,3,5	1,2,3,5
-5,0,1,97	1,97	1,97

6. Скриншоты

Введите размер массива: 4 arr[0] = -5 arr[1] = 0 arr[2] = 1 arr[3] = 97 Выберете действие: 0: Завершить программу 1: Пересоздать массив 2: Вставить новый элемент 3: Удалить элемент 4: Обработка данных 5: Вывод данных >>> 4 Выберете действие: 0: Завершить программу 1: Пересоздать массив 2: Вставить новый элемент 3: Удалить элемент 4: Обработка данных 5: Вывод данных >>> 5 Size: 4 Count: 2 arr[0]: 1 arr[1]: 97	Выберете действие: 0: Завершить программу 1: Пересоздать массив 2: Вставить новый элемент 3: Удалить элемент 4: Обработка данных 5: Вывод данных >>> 5 Size: 8 Count: 4 arr[0]: 1 arr[1]: 2 arr[2]: 3 arr[3]: 5 Выберете действие: 0: Завершить программу 1: Пересоздать массив 2: Вставить новый элемент 3: Удалить элемент 4: Обработка данных 5: Вывод данных >>> 1
--	--

7. Выводы

В ходе выполнения данной работы на примере программы, выполняющей поиск простых чисел и удаление остальных из массива, разбивать проект на несколько файлов, а также работать со структурой и выделять/освобождать для нее память.