

Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №12 «Компьютерные системы и технологии»



ОТЧЕТ

**О выполнении лабораторной работы №3
«Работа с массивами данных.»**

Студент: Титов Д.И.

Группа: Б22 - 505

Преподаватель: Вавренюк А.Б

Москва — 2022

1. Формулировка индивидуального задания

Вариант №30.

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

При выполнении данной лабораторной работы использовался встроенный тип данных `int` предназначенный для работы с целыми числами и указатель типа `int`. Также использовались заголовочные файлы для формирования программы как сборного проекта из нескольких файлов

3. Описание использованного алгоритма

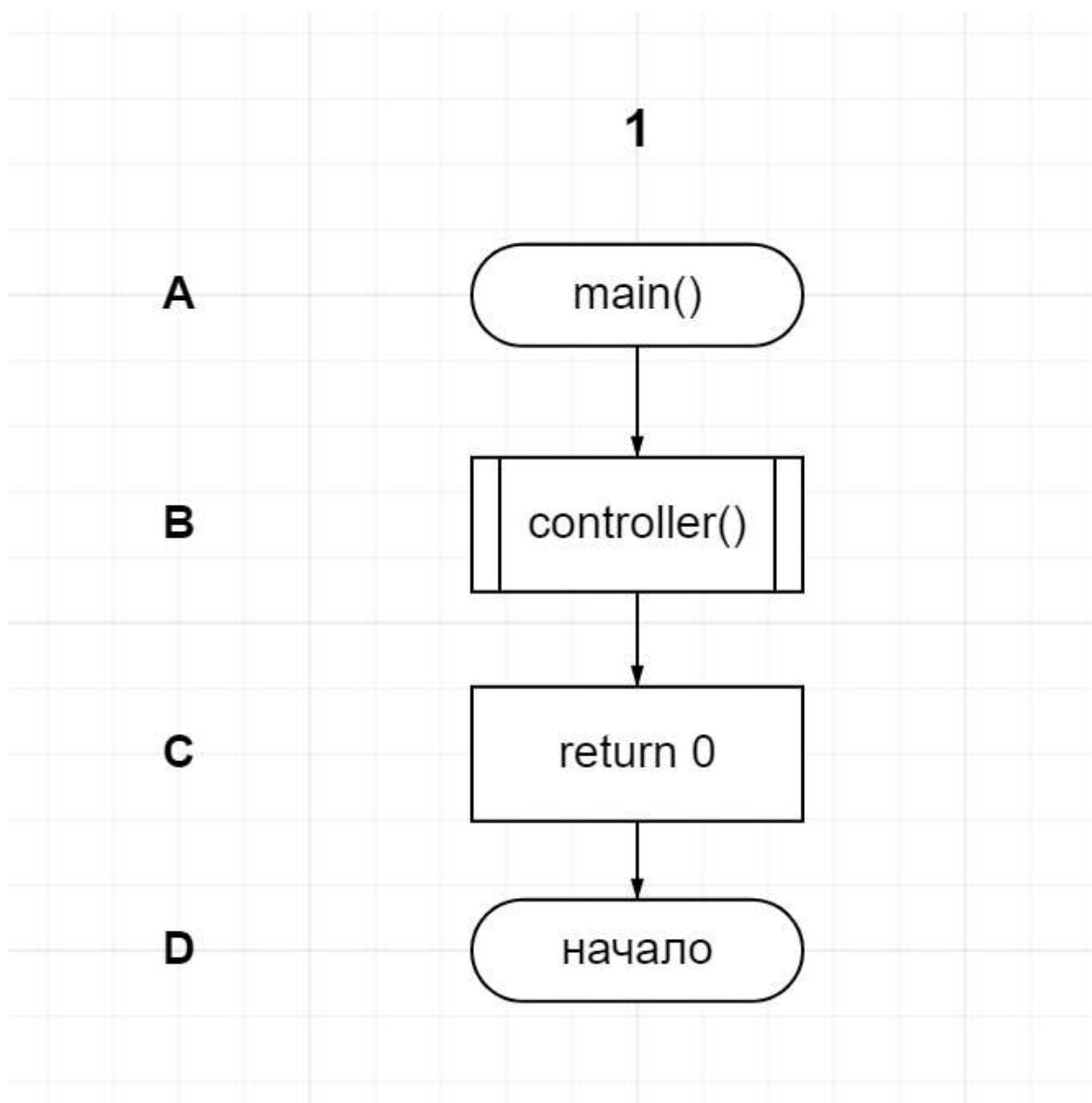


Рис. 1: Блоксхема алгоритма работы функции `main()`

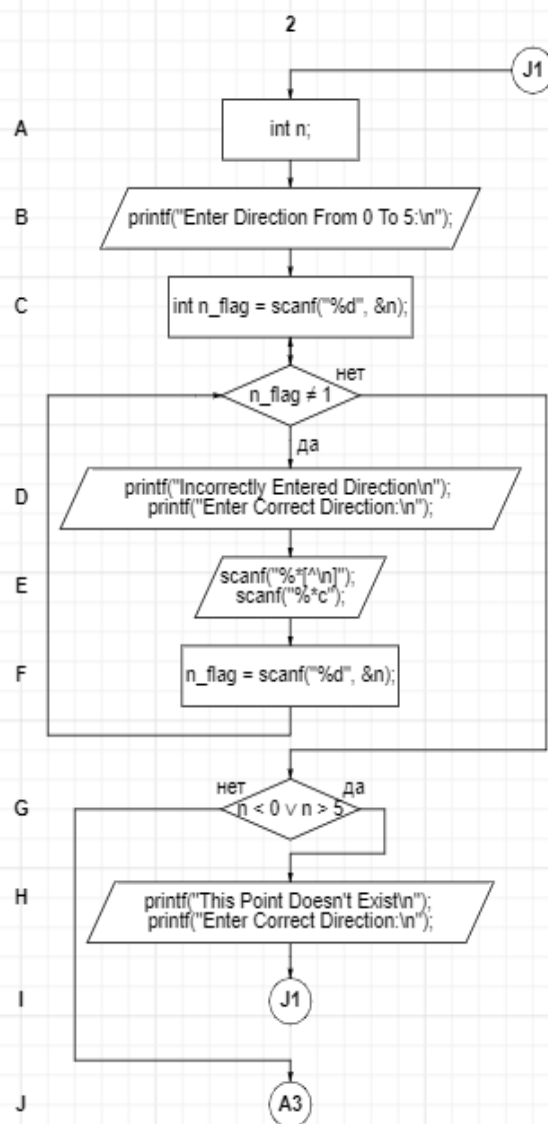
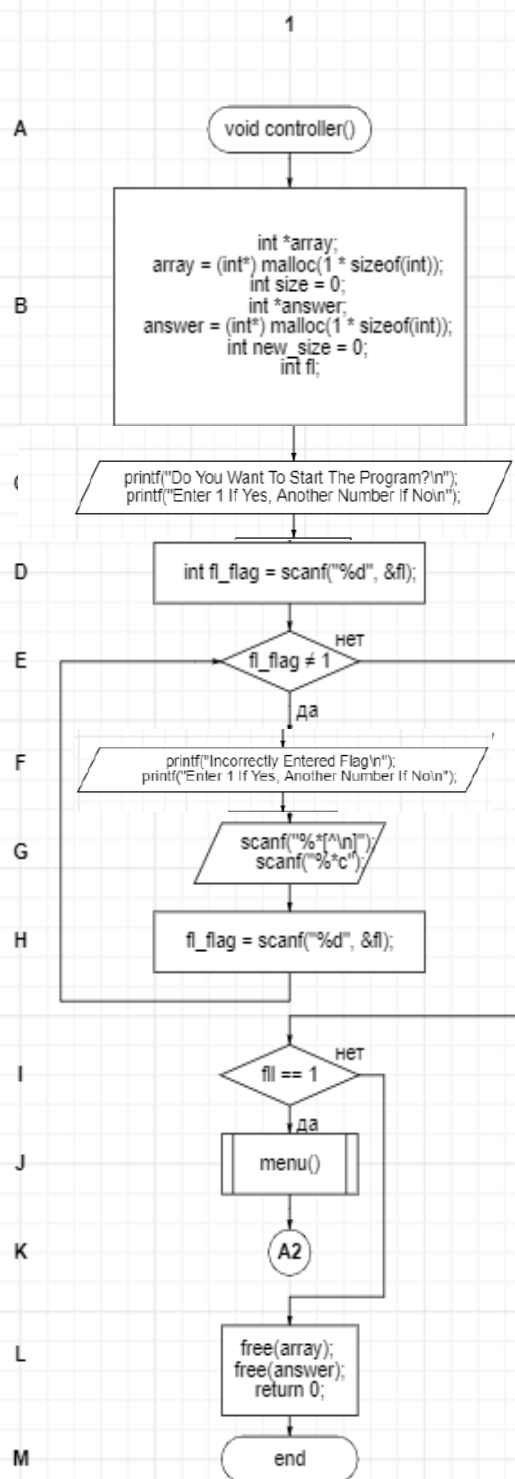
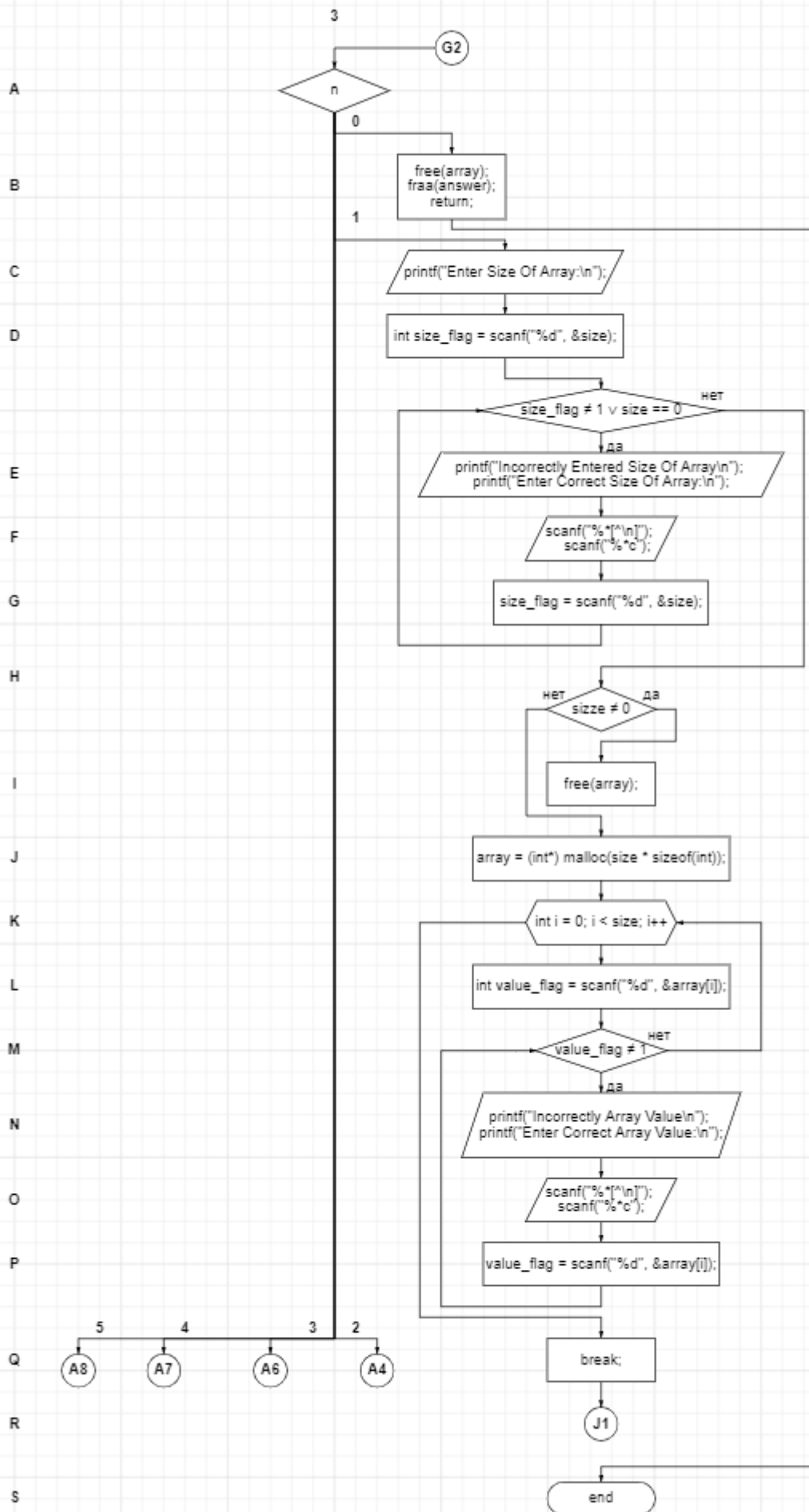
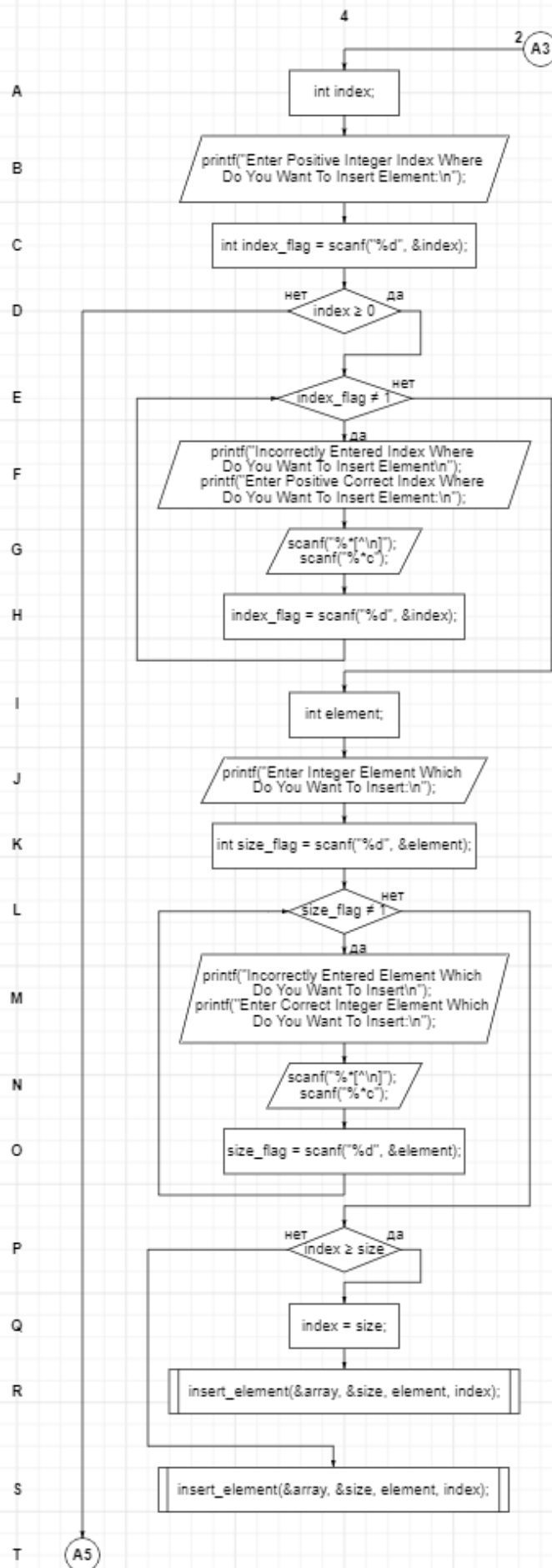
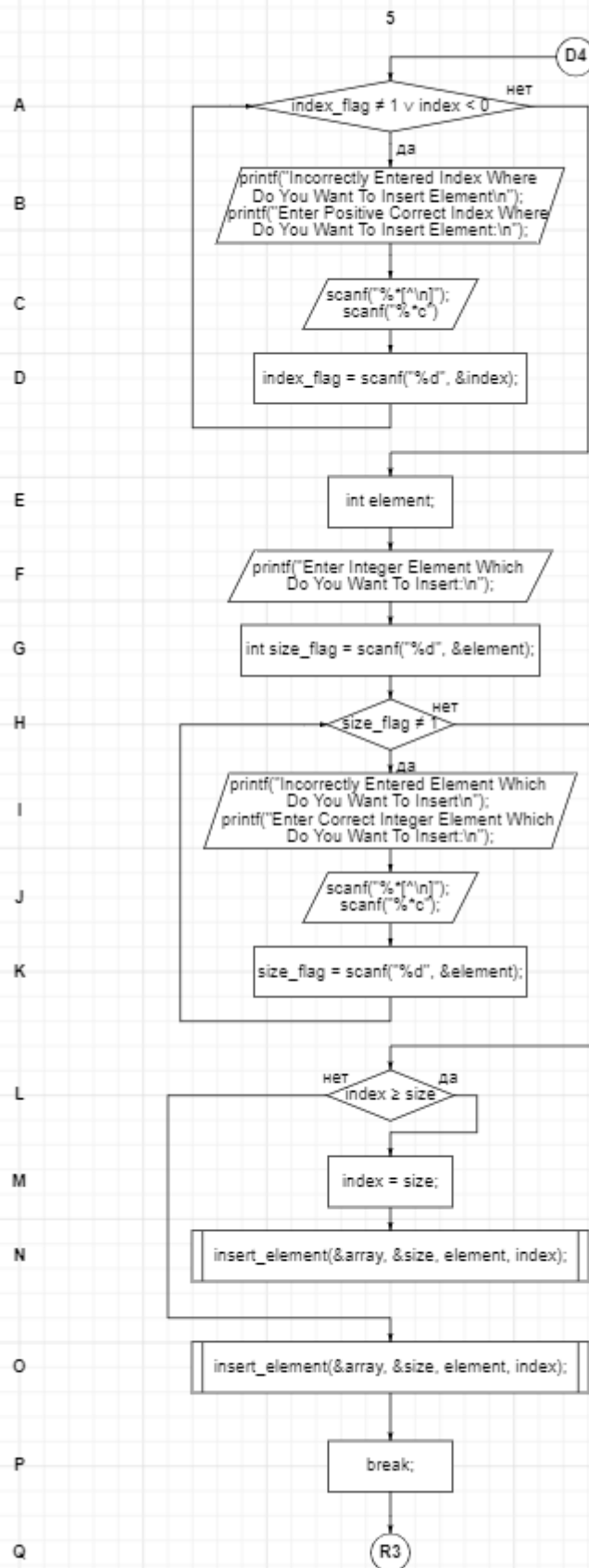
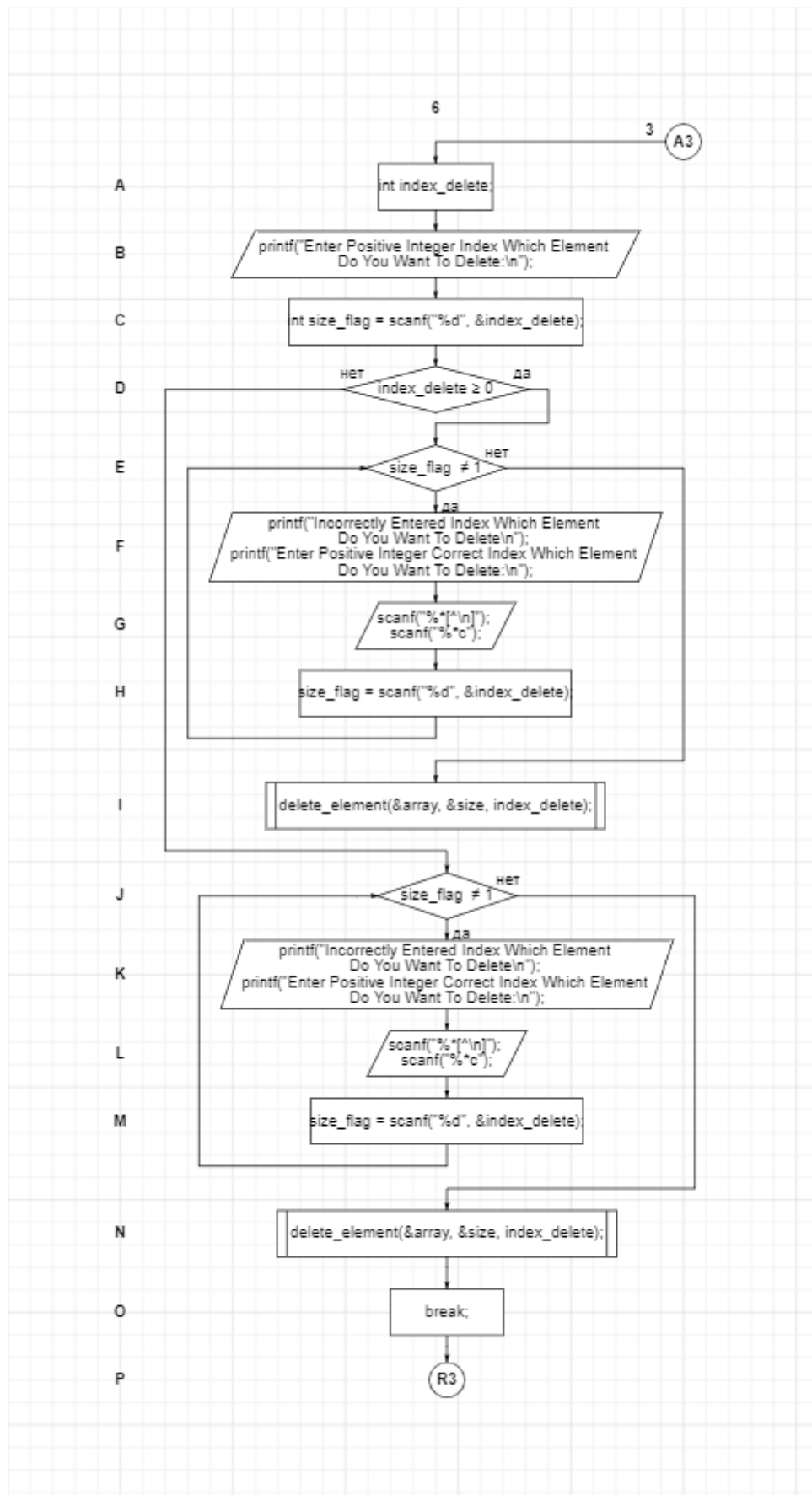


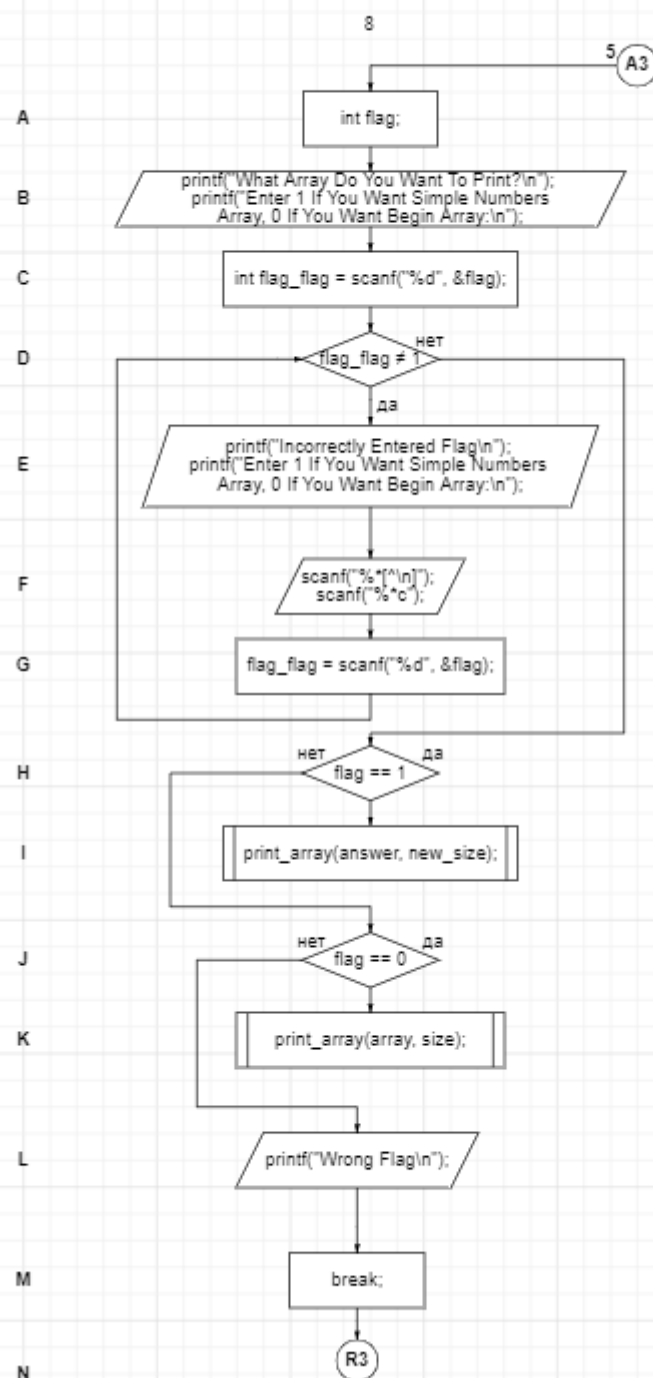
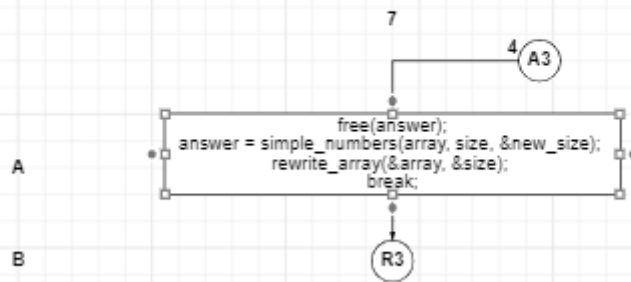
Рис. 2(стр 3 — 8) : Блоксхема алгоритма работы функции controller()











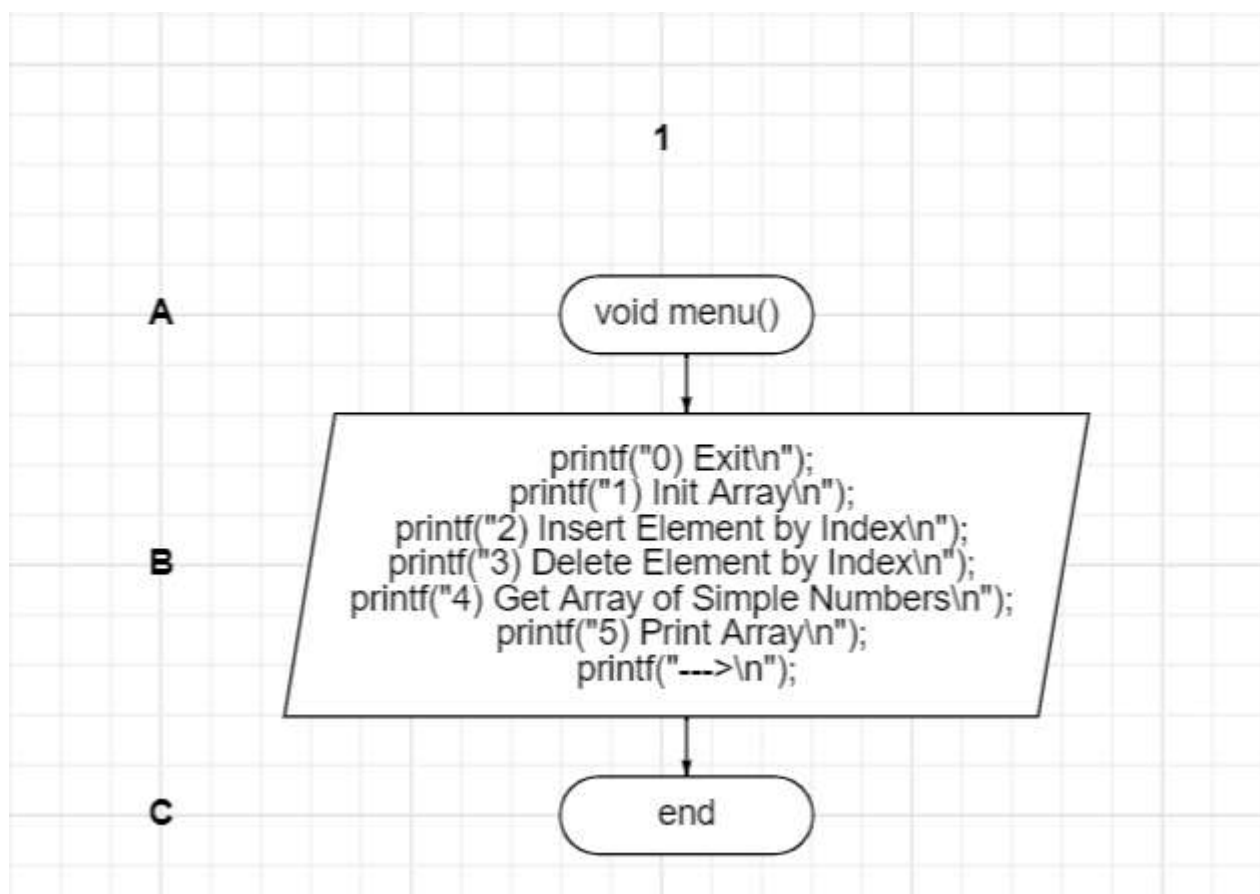


Рис. 3: Блоксхема алгоритма работы функции `menu()`

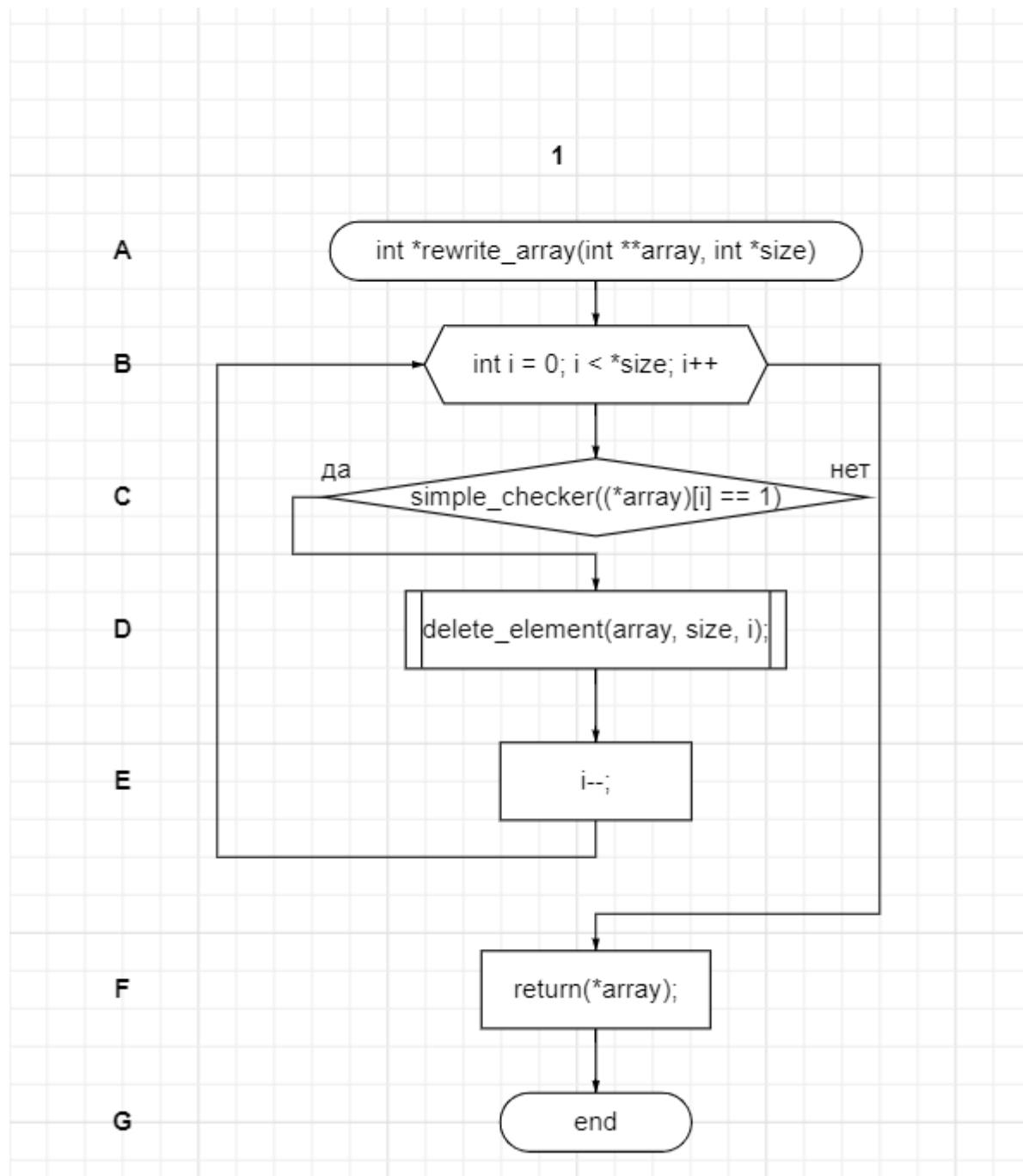


Рис. 4: Блоксхема алгоритма работы функции `*rewrite_array()`

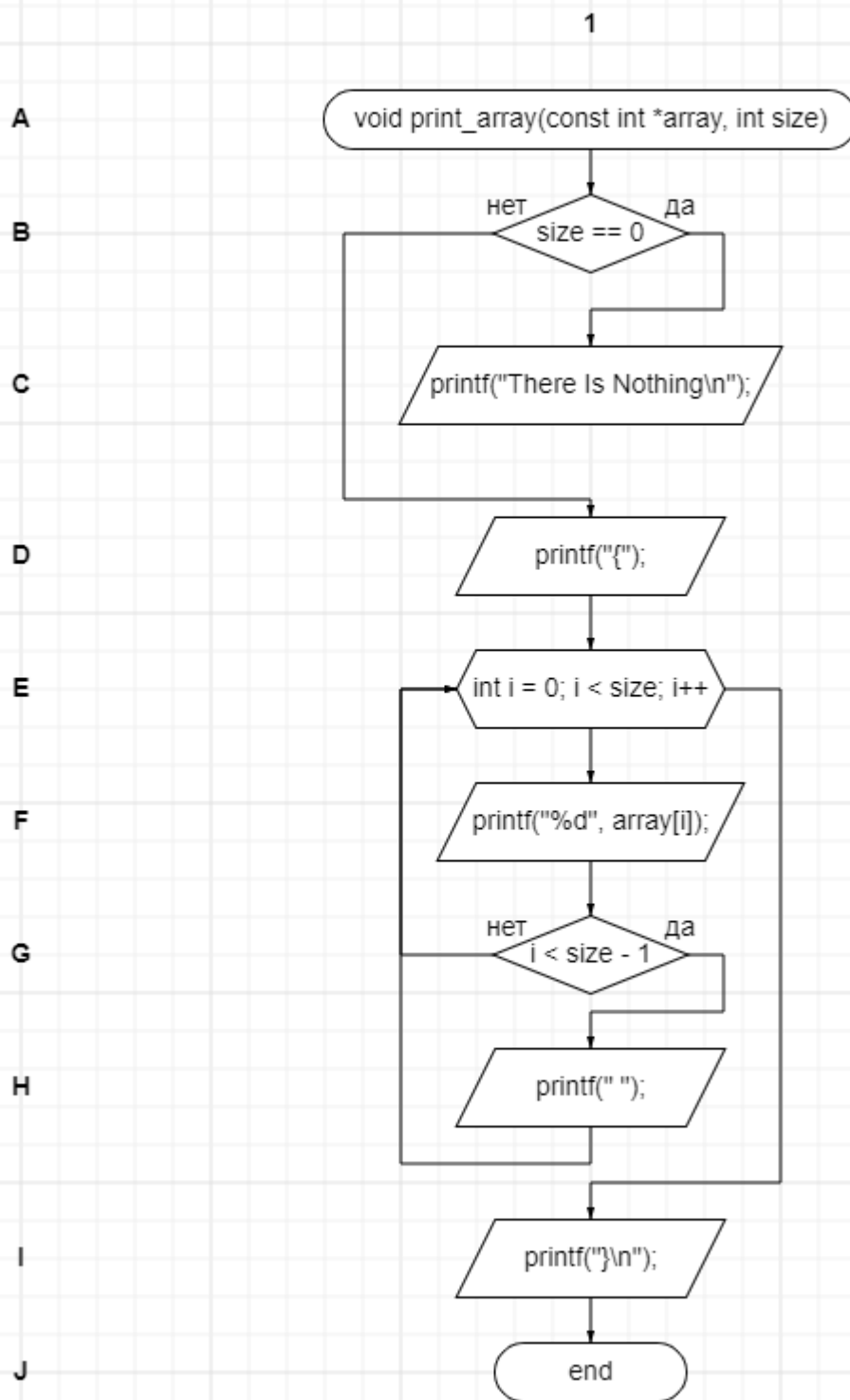


Рис. 5: Блоксхема алгоритма работы функции print_array()

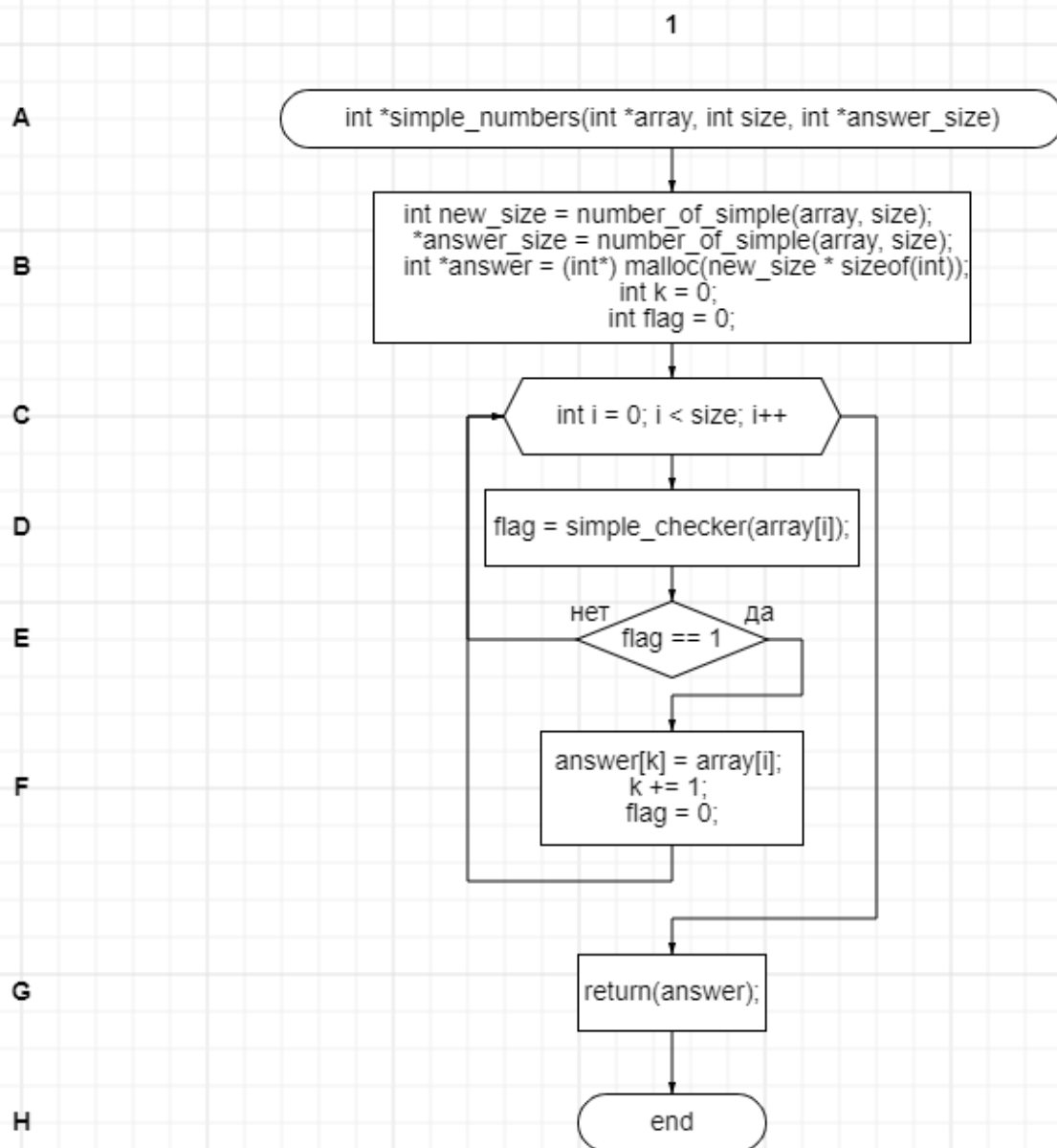


Рис. 6: Блоксхема алгоритма работы функции *simple_numbers()

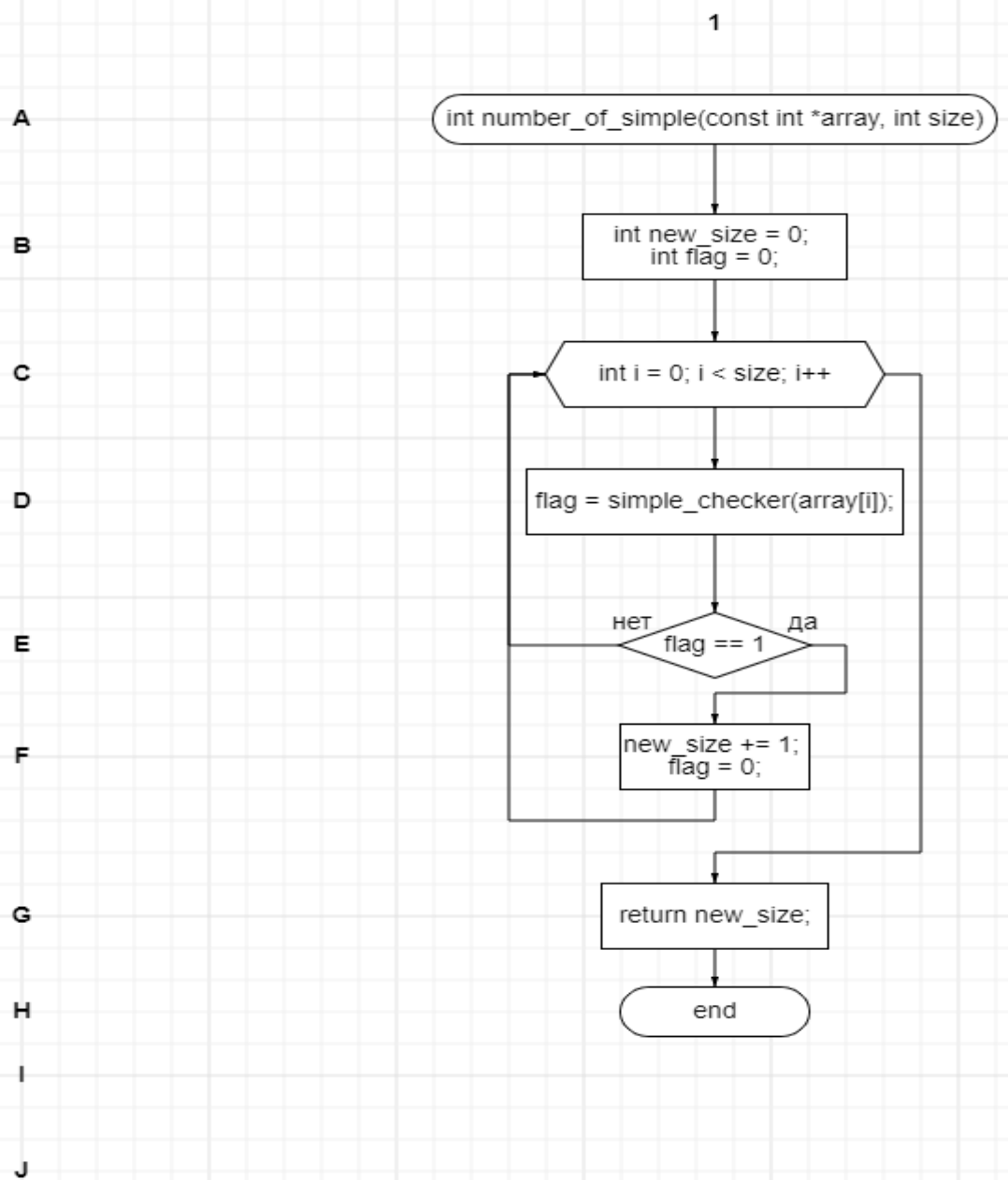


Рис. 7: Блоксхема алгоритма работы функции number_of_simple()

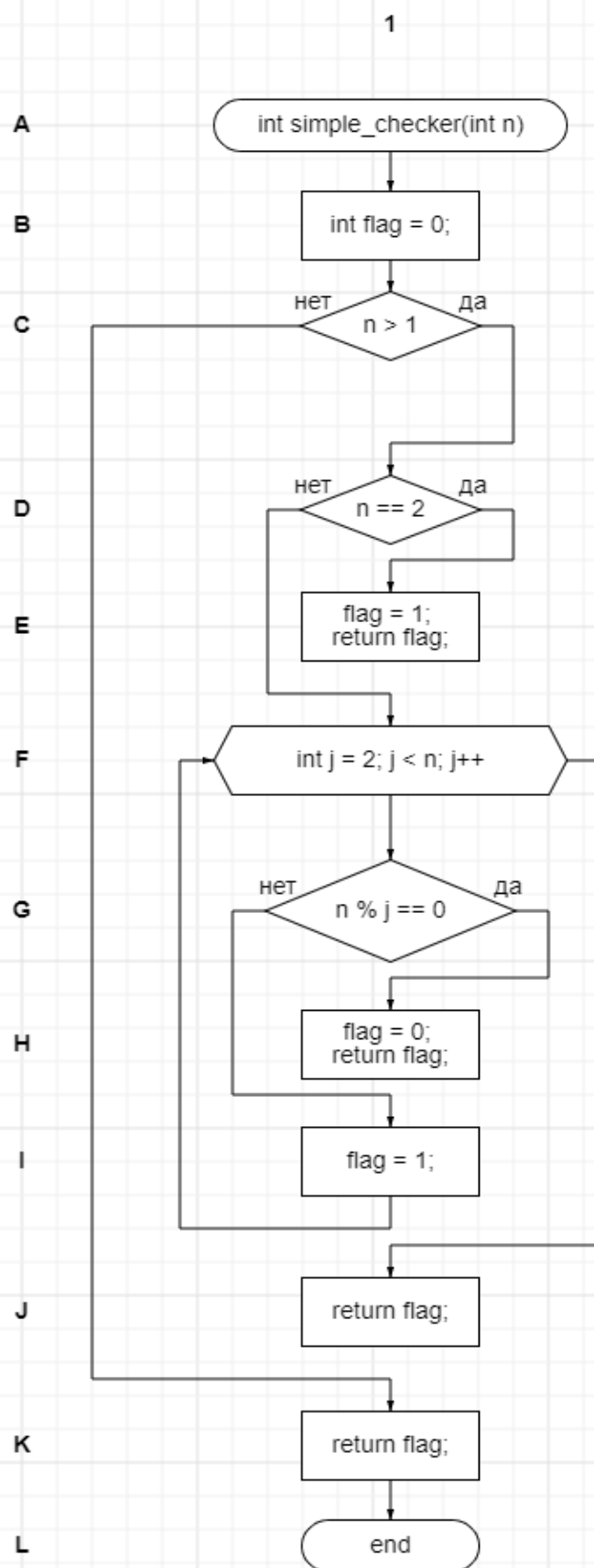


Рис. 8: Блоксхема алгоритма работы функции simple_checker()

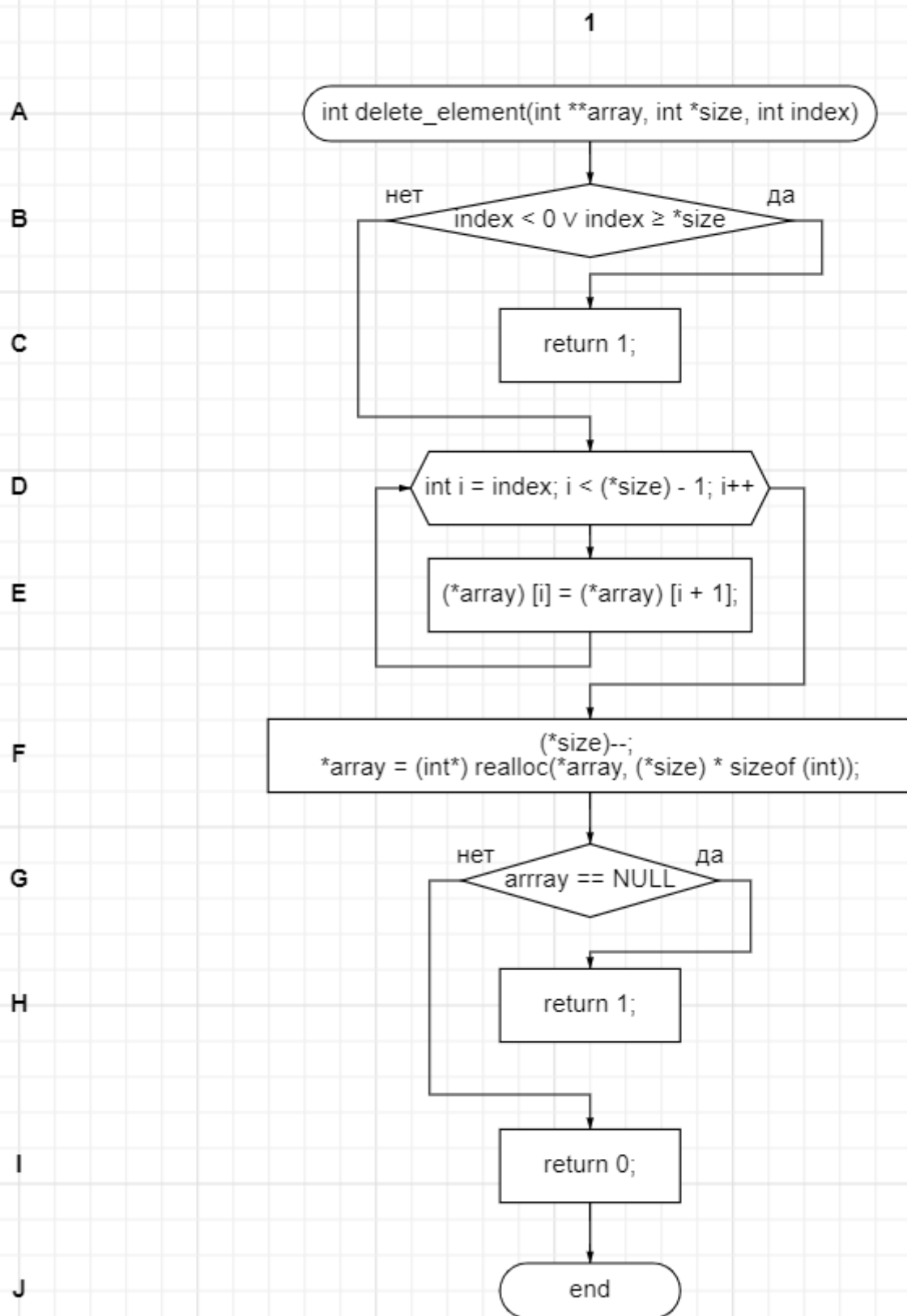


Рис. 8: Блоксхема алгоритма работы функции delete_element()

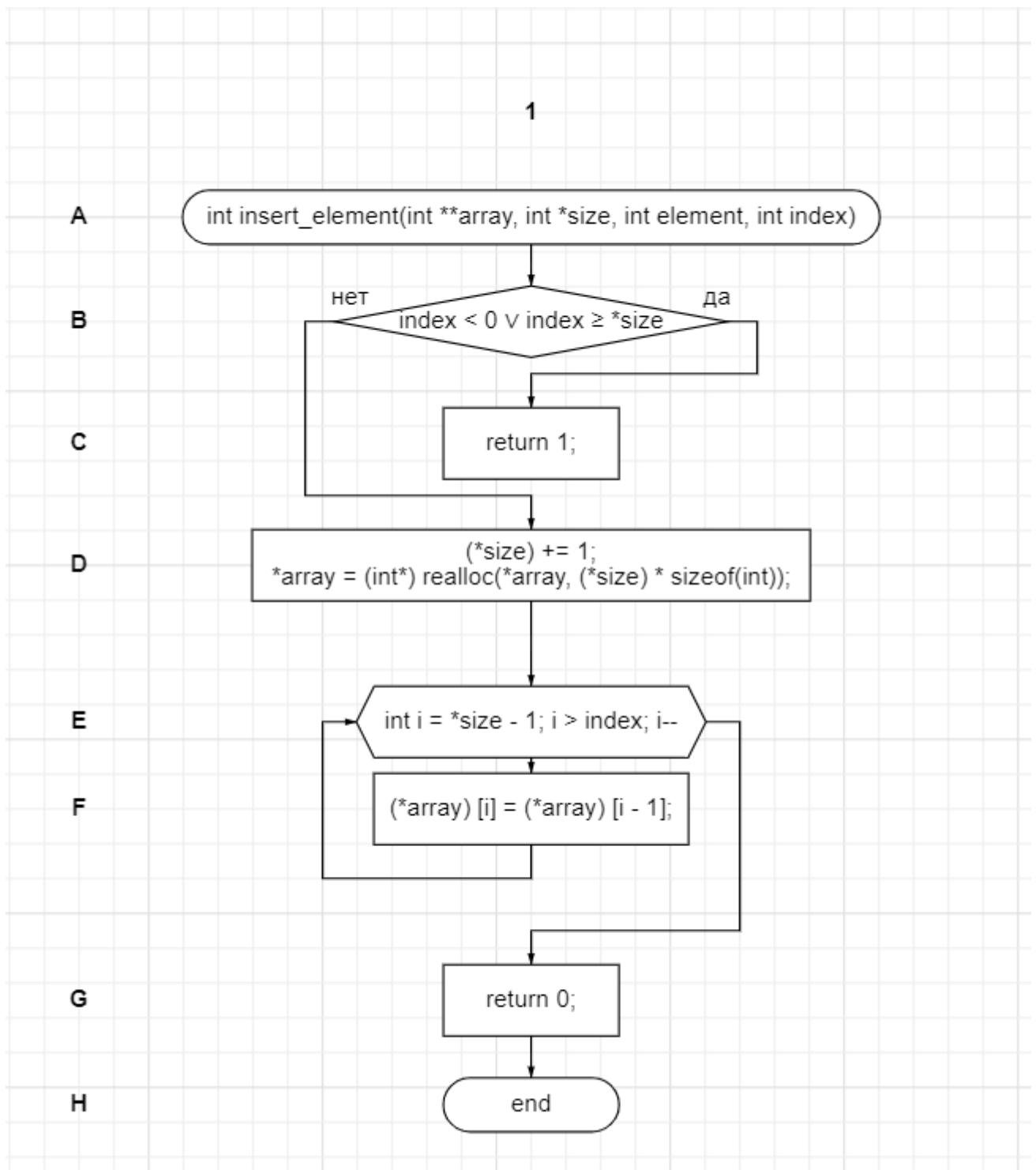


Рис. 9: Блоксхема алгоритма работы функции `insert_element()`

4. Исходные коды разработанных программ

```
#include <stdio.h>
#include "controller.h"
#include <stdlib.h>

int main()
{
    controller();
}
```

Листинг 1: исходные коды программы program (файл: lab3main.c)

```
#ifndef LAB3_ARRAY_H
#define LAB3_ARRAY_H

int insert_element(int **, int *, int, int);
int delete_element(int **, int *, int);

int simple_checker(int);
int number_of_simple(const int *, int);
int *simple_numbers(int *, int, int *);
int *rewrite_array(int **, int *);

#endif //LAB3_ARRAY_H
```

Листинг 2: исходные коды программы program (файл: array.h)

```
#ifndef LAB3_VIEW_H
#define LAB3_VIEW_H

void menu();
void print_array(const int *, int);

#endif //LAB3_VIEW_H
```

Листинг 3: исходные коды программы program (файл: view.h)

```
#ifndef LAB3_CONTROLLER_H
#define LAB3_CONTROLLER_H

void controller();

#endif //LAB3_CONTROLLER_H
```

Листинг 4: исходные коды программы program (файл: controller.h)

```

#include "array.h"
#include <stdlib.h>

int insert_element(int **array, int *size, int element, int index) {
    if (index < 0 || index > *size) {
        return 1;
    }
    *size += 1;
    *array = (int*) realloc(*array, (*size) * sizeof(int));
    for (int i = *size - 1; i > index; i--) {
        (*array)[i] = (*array)[i - 1];
    }
    (*array)[index] = element;
    if (array == NULL) {
        return 1;
    }
    return 0;
}

int delete_element(int **array, int *size, int index) {
    if (index < 0 || index >= *size) {
        return 1;
    }
    for (int i = index; i < (*size) - 1; i++) {
        (*array)[i] = (*array)[i + 1];
    }
    (*size)--;
    *array = (int*) realloc(*array, (*size) * sizeof(int));
    if (array == NULL) {
        return 1;
    }
    return 0;
}

int simple_checker(int n) {
    int flag = 0;
    if (n > 1) {
        if (n == 2) {
            flag = 1;
            return flag;
        }
        for (int j = 2; j < n; j++) {
            if (n % j == 0) {
                flag = 0;
                return flag;
            } else {
                flag = 1;
            }
        }
        return flag;
    }
    return flag;
}

int number_of_simple(const int *array, int size) {
    int new_size = 0;
    int flag = 0;
    for (int i = 0; i < size; i++) {
        flag = simple_checker(array[i]);
        if (flag == 1) {
            new_size += 1;
            flag = 0;
        }
    }
    return new_size;
}

int *simple_numbers(int *array, int size, int *answer_size) {
    int new_size = number_of_simple(array, size);
    *answer_size = number_of_simple(array, size);
    int *answer = (int*) malloc(new_size * sizeof(int));
    int k = 0;
    int flag = 0;
    for (int i = 0; i < size; i++) {
        flag = simple_checker(array[i]);
        if (flag == 1) {
            answer[k] = array[i];
            k += 1;
            flag = 0;
        }
    }
    return answer;
}

```

Листинг 5: исходные коды программы program (файл: array.c)

```

int *rewrite_array(int **array, int *size) {
    for (int i = 0; i < *size; i++) {
        if (simple_checker((*array)[i]) == 1) {
            delete_element(array, size, i);
            i--;
        }
    }
    return (*array);
}

```

Листинг 6: исходные коды программы program (файл: array.c)

```

#include "view.h"
#include <stdio.h>

void menu() {
    printf("0) Exit\n");
    printf("1) Init Array\n");
    printf("2) Insert Element by Index\n");
    printf("3) Delete Element by Index\n");
    printf("4) Get Array of Simple Numbers\n");
    printf("5) Print Array\n");
    printf("---->\n");
}

void print_array(const int *array, int size) {
    if (size == 0) {
        printf("There Is Nothing\n");
    } else {
        printf("{");
        for (int i = 0; i < size; i++) {
            printf("%d", array[i]);
            if (i < size - 1) {
                printf(" ");
            }
        }
        printf("}\n");
    }
}

```

Листинг 7: исходные коды программы program (файл: view.c)

```

#include <stdio.h>
#include "controller.h"
#include "view.h"
#include "array.h"
#include <stdlib.h>

void controller() {
    int *array;
    array = (int*) malloc(1 * sizeof(int));
    int size = 0;
    int *answer;
    answer = (int*) malloc(1 * sizeof(int));
    int new_size = 0;
    int fl;
    printf("Do You Want To Start The Program?\n");
    printf("Enter 1 If Yes, Another Number If No\n");
    int fl_flag = scanf("%d", &fl);
    while (fl_flag != 1) {
        printf("Incorrectly Entered Flag\n");
        printf("Enter 1 If Yes, Another Number If No\n");
        scanf("%*[\n]");
        scanf("%*c");
        fl_flag = scanf("%d", &fl);
    }
    while (fl == 1) {
        menu();
        int n;
        printf("Enter Direction From 0 To 5:\n");
        int n_flag = scanf("%d", &n);
        while (n_flag != 1) {
            printf("Incorrectly Entered Direction\n");
            printf("Enter Correct Direction:\n");
            scanf("%*[\n]");
            scanf("%*c");
            n_flag = scanf("%d", &n);
        }
        if (n < 0 || n > 5) {
            printf("This Point Doesn't Exist\n");
            printf("Enter Correct Direction:\n");
        } else {
            switch (n) {
                case 0: {
                    free(array);
                    free(answer);
                    return;
                }
                case 1: {
                    printf("Enter Size Of Array:\n");
                    int size_flag = scanf("%d", &size);
                    while (size_flag != 1 || size == 0) {
                        printf("Incorrectly Entered Size Of Array\n");
                        printf("Enter Correct Size Of Array:\n");
                        scanf("%*[\n]");
                        scanf("%*c");
                        size_flag = scanf("%d", &size);
                    }
                    if (size != 0) {
                        free(array);
                    }
                    array = (int*) malloc(size * sizeof(int));
                    for (int i = 0; i < size; i++) {
                        int value_flag = scanf("%d", &array[i]);
                        while (value_flag != 1) {
                            printf("Incorrectly Array Value\n");
                            printf("Enter Correct Array Value:\n");
                            scanf("%*[\n]");
                            scanf("%*c");
                            value_flag = scanf("%d", &array[i]);
                        }
                    }
                    break;
                }
                case 2: {
                    int index;
                    printf("Enter Positive Integer Index Where Do You Want To Insert Element:\n");
                    int index_flag = scanf("%d", &index);
                    if (index >= 0) {
                        while (index_flag != 1) {
                            printf("Incorrectly Entered Index Where Do You Want To Insert Element\n");
                            printf("Enter Positive Correct Index Where Do You Want To Insert Element:\n");
                            scanf("%*[\n]");
                            scanf("%*c");
                            index_flag = scanf("%d", &index);
                        }
                        int element;

```

Листинг 8: исходные коды программы program (файл: controller.c)

```

        int element;
        printf("Enter Integer Element Which Do You Want To Insert:\n");
        int size_flag = scanf("%d", &element);
        while (size_flag != 1) {
            printf("Incorrectly Entered Element Which Do You Want To Insert:\n");
            printf("Enter Correct Integer Element Which Do You Want To Insert:\n");
            scanf("%i^\n");
            scanf("%i^\n");
            size_flag = scanf("%d", &element);
        }
        if (index >= size) {
            index = size;
            insert_element(&array, &size, element, index);
        } else {
            insert_element(&array, &size, element, index);
        }
    } else {
        while (index_flag != 1 || index < 0) {
            printf("Incorrectly Entered Index Where Do You Want To Insert Element:\n");
            printf("Enter Positive Correct Index Where Do You Want To Insert Element:\n");
            scanf("%i^\n");
            scanf("%i^\n");
            index_flag = scanf("%d", &index);
        }
        int element;
        printf("Enter Integer Element Which Do You Want To Insert:\n");
        int size_flag = scanf("%d", &element);
        while (size_flag != 1) {
            printf("Incorrectly Entered Element Which Do You Want To Insert:\n");
            printf("Enter Correct Integer Element Which Do You Want To Insert:\n");
            scanf("%i^\n");
            scanf("%i^\n");
            size_flag = scanf("%d", &element);
        }
        if (index >= size) {
            index = size;
            insert_element(&array, &size, element, index);
        } else {
            insert_element(&array, &size, element, index);
        }
    }
    break;
}
case 3: {
    int index_delete;
    printf("Enter Positive Integer Index Which Element Do You Want To Delete:\n");
    int size_flag = scanf("%d", &index_delete);
    if (index_delete >= 0) {
        while (size_flag != 1) {
            printf("Incorrectly Entered Index Which Element Do You Want To Delete:\n");
            printf("Enter Positive Integer Correct Index Which Element Do You Want To Delete:\n");
            scanf("%i^\n");
            scanf("%i^\n");
            size_flag = scanf("%d", &index_delete);
        }
        delete_element(&array, &size, index_delete);
    } else {
        while (size_flag != 1 || index_delete < 0) {
            printf("Incorrectly Entered Index Which Element Do You Want To Delete:\n");
            printf("Enter Positive Integer Correct Index Which Element Do You Want To Delete:\n");
            scanf("%i^\n");
            scanf("%i^\n");
            size_flag = scanf("%d", &index_delete);
        }
        delete_element(&array, &size, index_delete);
    }
    break;
}
case 4: {
    free(answer);
    answer = simple_numbers(array, size, &new_size);
    rewrite_array(&array, &size);
    break;
}
case 5: {
    int flag;
    printf("What Array Do You Want To Print?\n");
    printf("Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:\n");
    int flag_flag = scanf("%d", &flag);
    while (flag_flag != 1) {
        printf("Incorrectly Entered Flag\n");
        printf("Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:\n");
        scanf("%i^\n");
        scanf("%i^\n");
        flag_flag = scanf("%d", &flag);
    }
}
default: {
    break;
}
}
}
free(array);
free(answer);
}

```

Листинг 9: исходные коды программы program (файл: controller.c)

```

    while (flag_flag != 1) {
        printf("Incorrectly Entered Flag\n");
        printf("Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:\n");
        scanf("%i^\n");
        scanf("%i^\n");
        flag_flag = scanf("%d", &flag);
    }
    if (flag == 1) {
        print_array(answer, new_size);
        //free(answer);
    } else if (flag == 0) {
        print_array(array, size);
    } else {
        printf("Wrong Flag\n");
    }
}
default: {
    break;
}
}
}
free(array);
free(answer);
}
}

```

Листинг 10: исходные коды программы program (файл: controller.c)

5. Описание тестовых примеров

Таблица 1: Тестовые примеры старт программы

Ввод пользователя	Ожидаемый ответ	Полученный Ответ
-	Do You Want To Start The Program? Enter 1 If Yes, Another Number If <u>No</u>	Do You Want To Start The Program? Enter 1 If Yes, Another Number If <u>No</u>
1	0) Exit 1) <u>Init</u> Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5: -	0) Exit 1) <u>Init</u> Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5: -
Любое число	Завершение программы	Завершение программы
Строка	<u>Incorrectly</u> Entered Flag Enter 1 If Yes, Another Number If <u>No</u>	<u>Incorrectly</u> Entered Flag Enter 1 If Yes, Another Number If <u>No</u>

В случае вызова строки в меню программа все также выдаст ошибку о некорректно введенных Данных, и запросит пользователя повторить ввод данных, но теперь правильным образом.

Ввод пользователя	Ожидаемый ответ	Полученный Ответ
1 (при выборе пункта меню) 3 (размер массива) 2 5 4	Enter Size Of Array: - 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:	Enter Size Of Array: - 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:
0 (при выборе пункта в меню)	Завершение программы	Завершение программы
2 (при выборе пункта в меню) 2 (индекс) 3 (число)	Enter Positive Integer Index Where Do You Want To Insert Element: - Enter Integer Element Which Do You Want To Insert: - 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:	Enter Positive Integer Index Where Do You Want To Insert Element: - Enter Integer Element Which Do You Want To Insert: - 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:
3 (при выборе пункта в меню) 2 (индекс)	Enter Positive Integer Index Which Element Do You Want To Delete: - 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5: -	Enter Positive Integer Index Which Element Do You Want To Delete: - 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5: -
4 (при выборе пункта в меню)	0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:	0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:
5 (при выборе пункта в меню) 1 если 0, то через / ответ	What Array Do You Want To Print? Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array: - {2 5} / {4} 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:	What Array Do You Want To Print? Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array: - {2 5} / {4} 0) Exit 1) Init Array 2) Insert Element by Index 3) Delete Element by Index 4) Get Array of Simple Numbers 5) Print Array ---> Enter Direction From 0 To 5:

Таблица 2: Тестовые примеры правильного взаимодействия с меню

6. Скриншоты

```
[titov.di@unix:~/Lab3]$ ./program
Do You Want To Start The Program?
Enter 1 If Yes, Another Number If No
1
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
1
Enter Size Of Array:
3
2
5
dfsgfd
Incorrectly Array Value
Enter Correct Array Value:
4
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
dsfg
Incorrectly Entered Direction
Enter Correct Direction:
2
Enter Positive Integer Index Where Do You Want To Insert Element:
3
Enter Integer Element Which Do You Want To Insert:
6
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
3
Enter Positive Integer Index Which Element Do You Want To Delete:
fdgs
Incorrectly Entered Index Which Element Do You Want To Delete
Enter Positive Integer Correct Index Which Element Do You Want To Delete:
1
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
fgs
Incorrectly Entered Direction
Enter Correct Direction:
4
```

Рис. 11 запущен тест 1

Рис. 10 запущен тест 1

```
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
5
What Array Do You Want To Print?
Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:
sdfg
Incorrectly Entered Flag
Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:
1
{2}
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
5
What Array Do You Want To Print?
Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:
0
{4 6}
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
0
[titov.di@unix:~/Lab3]$ ./
```



```
[titov.di@unix:~/Lab3]$ ./program
Do You Want To Start The Program?
Enter 1 If Yes, Another Number If No
fgfgsdfg
Incorrectly Entered Flag
Enter 1 If Yes, Another Number If No
3456

[titov.di@unix:~/Lab3]$
```

Рис. 12 запусен тест 3

Рис. 11 запусен тест 2

```
[titov.di@unix:~/Lab3]$ ./program
Do You Want To Start The Program?
Enter 1 If Yes, Another Number If No
1
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
sdfdsaf
Incorrectly Entered Direction
Enter Correct Direction:
1
Enter Size Of Array:
sdfg
Incorrectly Entered Size Of Array
Enter Correct Size Of Array:
2
sdafdsf
Incorrectly Array Value
Enter Correct Array Value:
3
fsdgds
Incorrectly Array Value
Enter Correct Array Value:
5
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
4
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
gfadf
Incorrectly Entered Direction
Enter Correct Direction:
5
What Array Do You Want To Print?
Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:
sdfdsf
Incorrectly Entered Flag
Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:
1
{3 5}
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
5
What Array Do You Want To Print?
Enter 1 If You Want Simple Numbers Array, 0 If You Want Begin Array:
0
There Is Nothing
0) Exit
1) Init Array
2) Insert Element by Index
3) Delete Element by Index
4) Get Array of Simple Numbers
5) Print Array
--->
Enter Direction From 0 To 5:
0

[titov.di@unix:~/Lab3]$
```

7. Выводы

В ходе выполнения данной работы на примере программы, выполняющей следующие функции:

- 1) Инициализация массива
- 2) Вставка нового элемента в массив по индексу
- 3) Удаление элемента массива по индексу
- 4) В исходной последовательности целых чисел найти те, которые являются простыми. Сформировать из данных чисел новую последовательность, удалив их из исходной
- 5) Вывод текущего состояния массива

Были изучены навыки работы с динамическими областями памяти, освоены навыки работы с .h и .c файлами при сборке их в консоли в единую программу, изучены возможности отладчика Valgrind, также изучены основы работы с массива и их индексами.