

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: Кукса И. В..

Группа: Б23-503

Преподаватель: Бабалова И. Ф.

Москва — 2023

1. Формулировка индивидуального задания

Программа должна реализовывать следующую функциональность:

1. Инициализация массива (заполнение массива в цикле).
2. Вставка нового элемента в массив по индексу.
3. Удаление элемента массива по индексу.
4. Обработка данных (в соответствии с индивидуальным заданием).
5. Вывод текущего состояния массива.

Индивидуальное задание

В исходной последовательности найти группы подряд стоящих одинаковых чисел и создать из них новую последовательность. Удалить найденные группы из исходной последовательности.

2. Описание использованных типов данных

Память под массивы обрабатываемых данных должна выделяться и освобождаться динамически, с использованием функций `calloc()`, `malloc()`, `realloc()` и `free()` из состава стандартной библиотеки.

Правила изменения размера выделенной под массив области памяти

Размер выделенной под массив области памяти должен изменяться при выполнении операций, приводящих к изменению количества элементов в массиве.

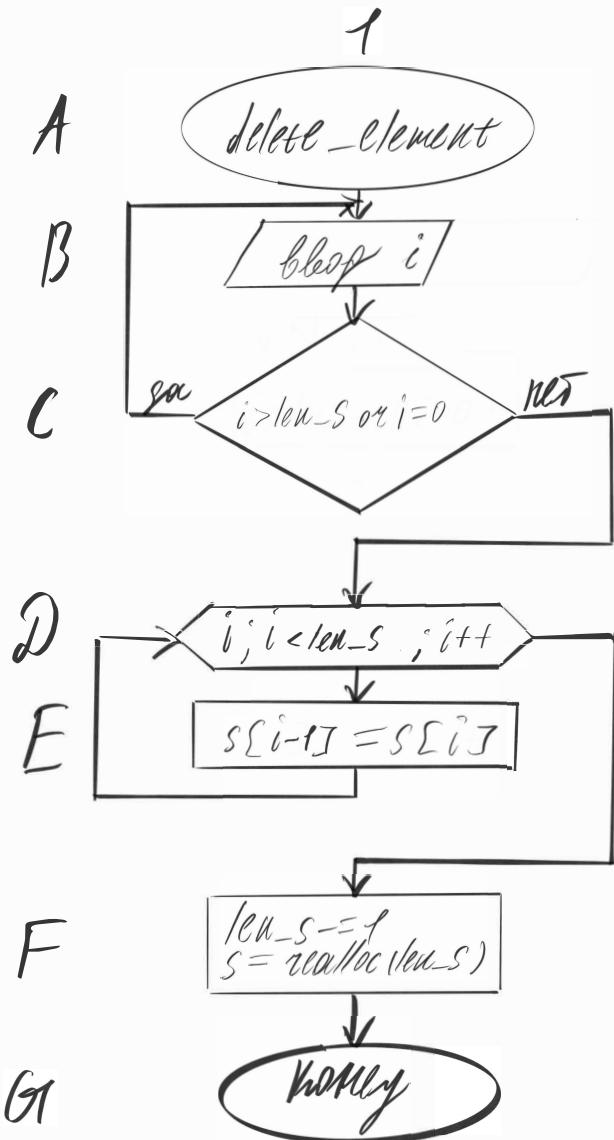
При заполнении элементами массива всей выделенной под него области памяти её размер должен удваиваться. При заполнении элементами массива выделенной под него области памяти менее, чем наполовину, её размер должен сокращаться вдвое.

В функциях вставки и удаления элементов массива должна быть предусмотрена соответствующая функциональность.

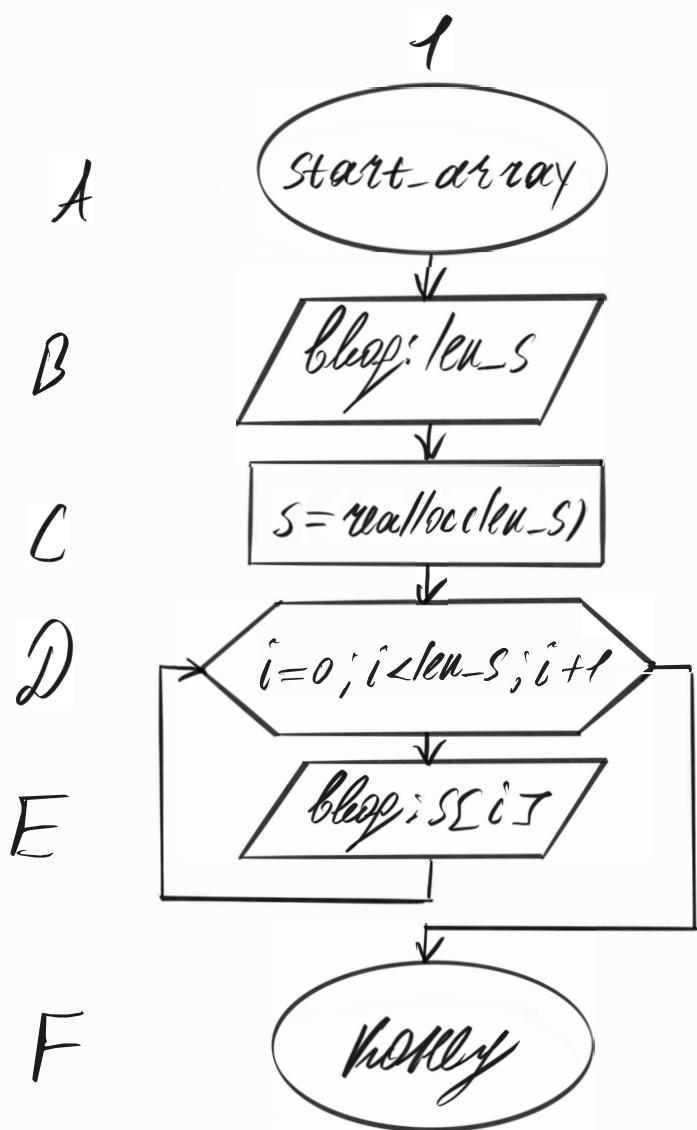
Исходные коды программы должны быть логичным образом разбиты на несколько файлов (необходимо использовать как *.c-файлы, так и *.h-файлы).

Программа должна штатным образом завершаться при выборе пользователем соответствующего пункта диалогового меню или при обнаружении EOF — конца файла (в UNIX-подобных ОС инициируется нажатием клавиш `Ctrl + D`, в Windows — `Ctrl + Z`).

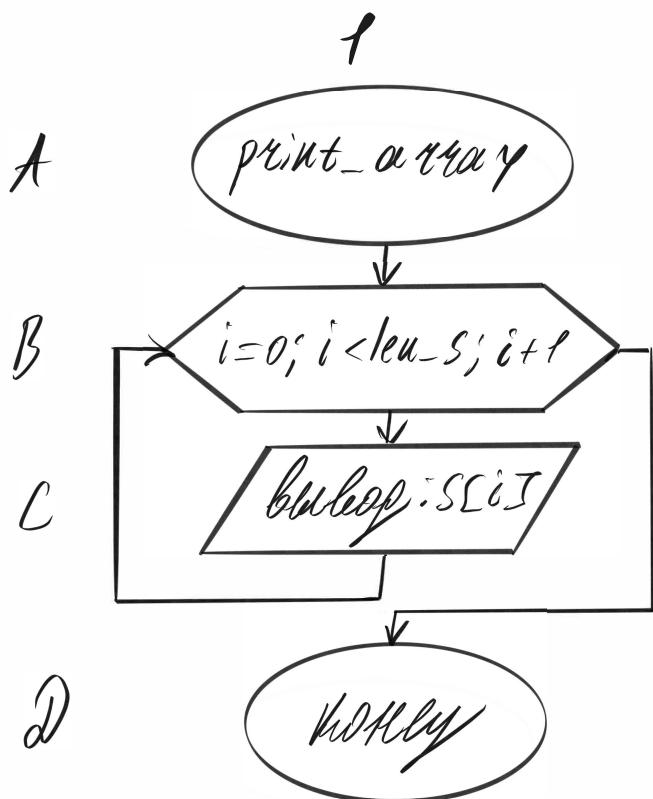
3. Описание использованного алгоритма



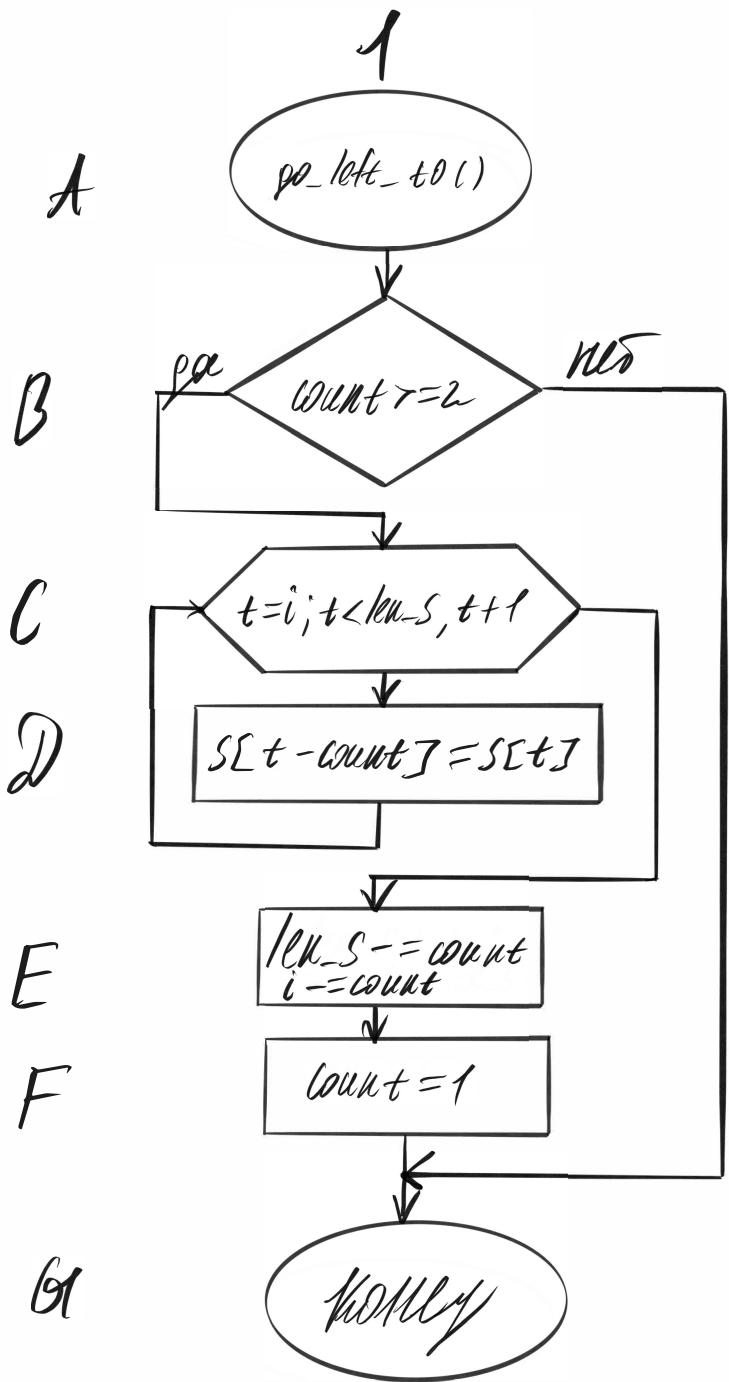
ис. 1: лок-схема функции `array.h`
(`delete_element`)



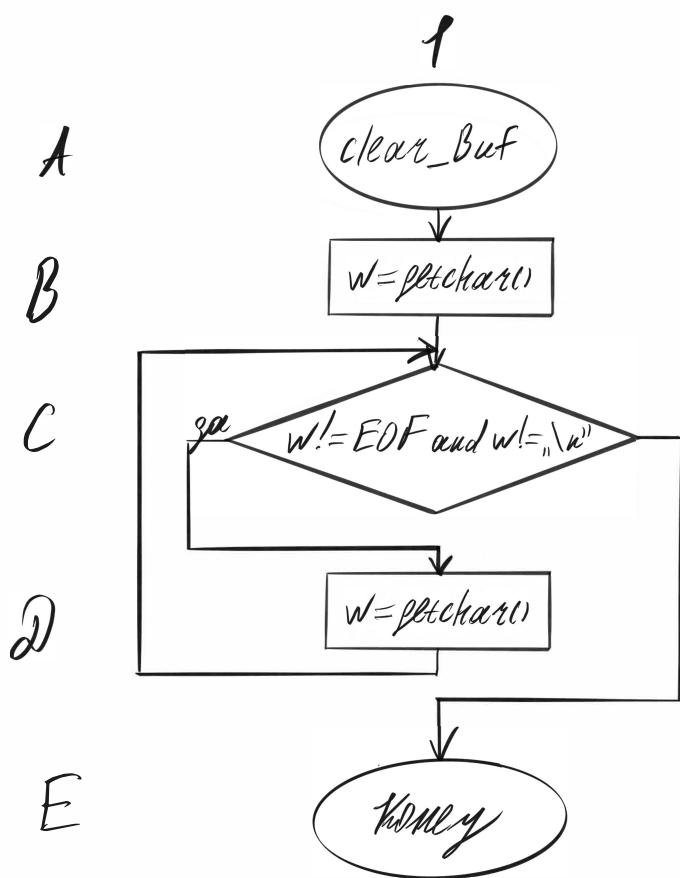
ис. 2: лок-схема функции `array.h`
(`start_array`)



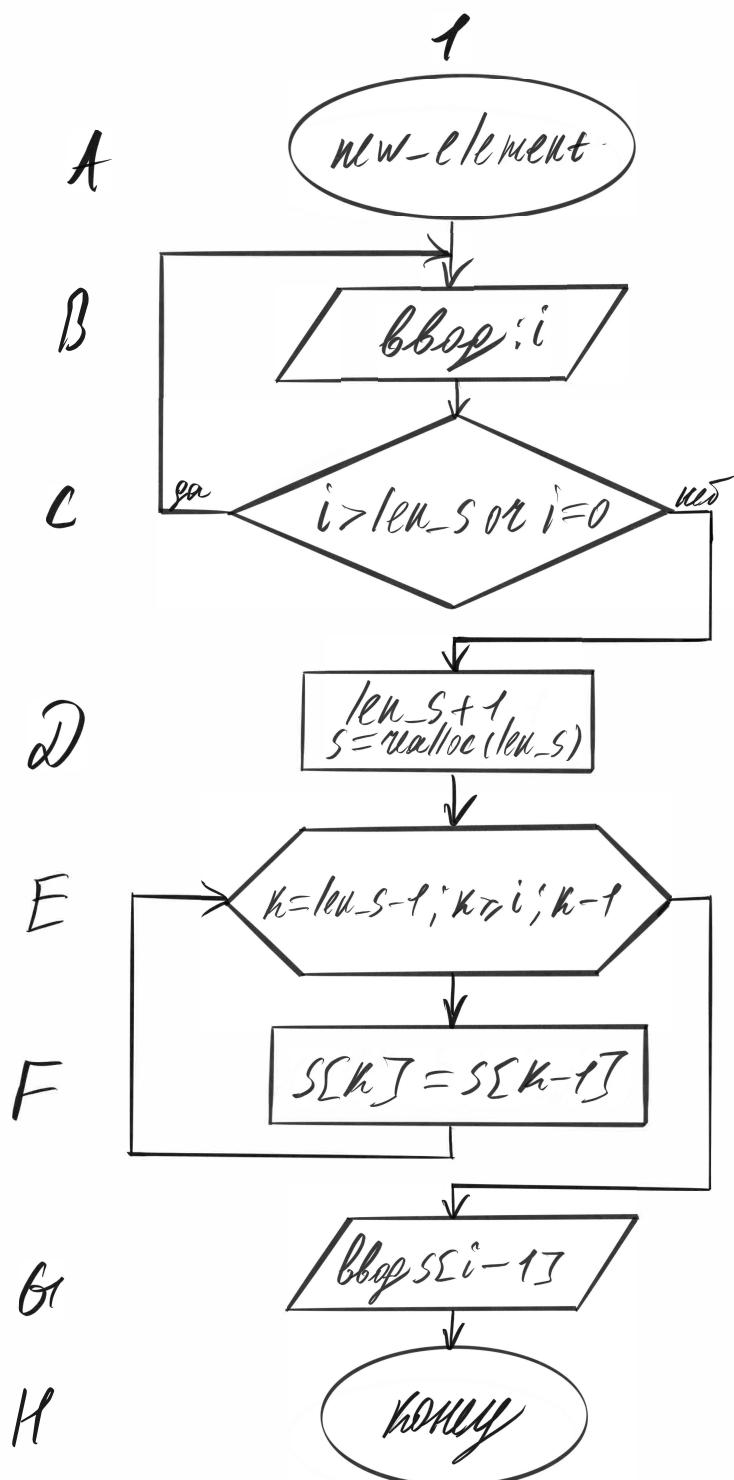
ис. 3: лок-схема функции array.h
(print_array)



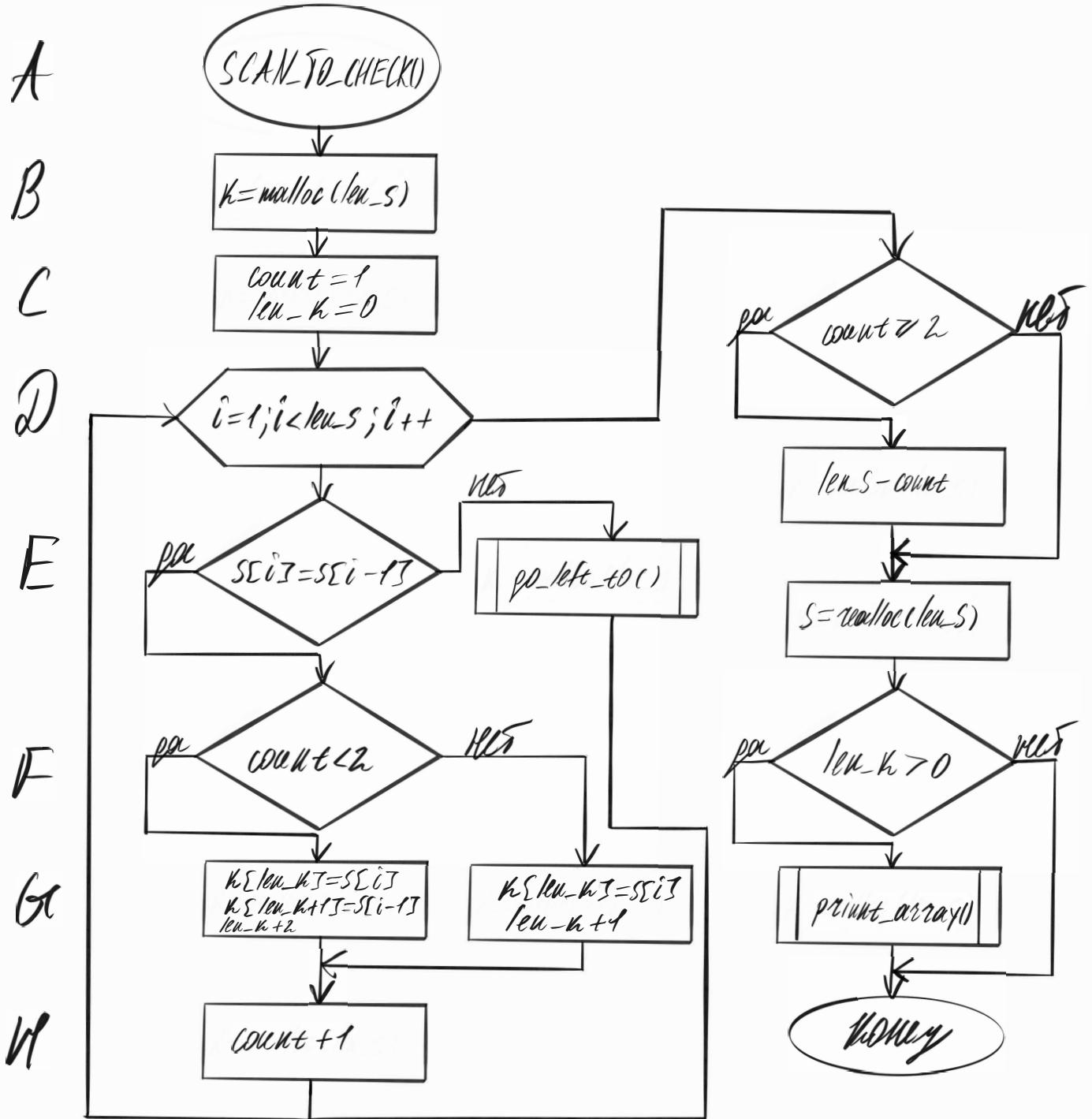
ис. 4: лок-схема функции array.h
(go_left_to)



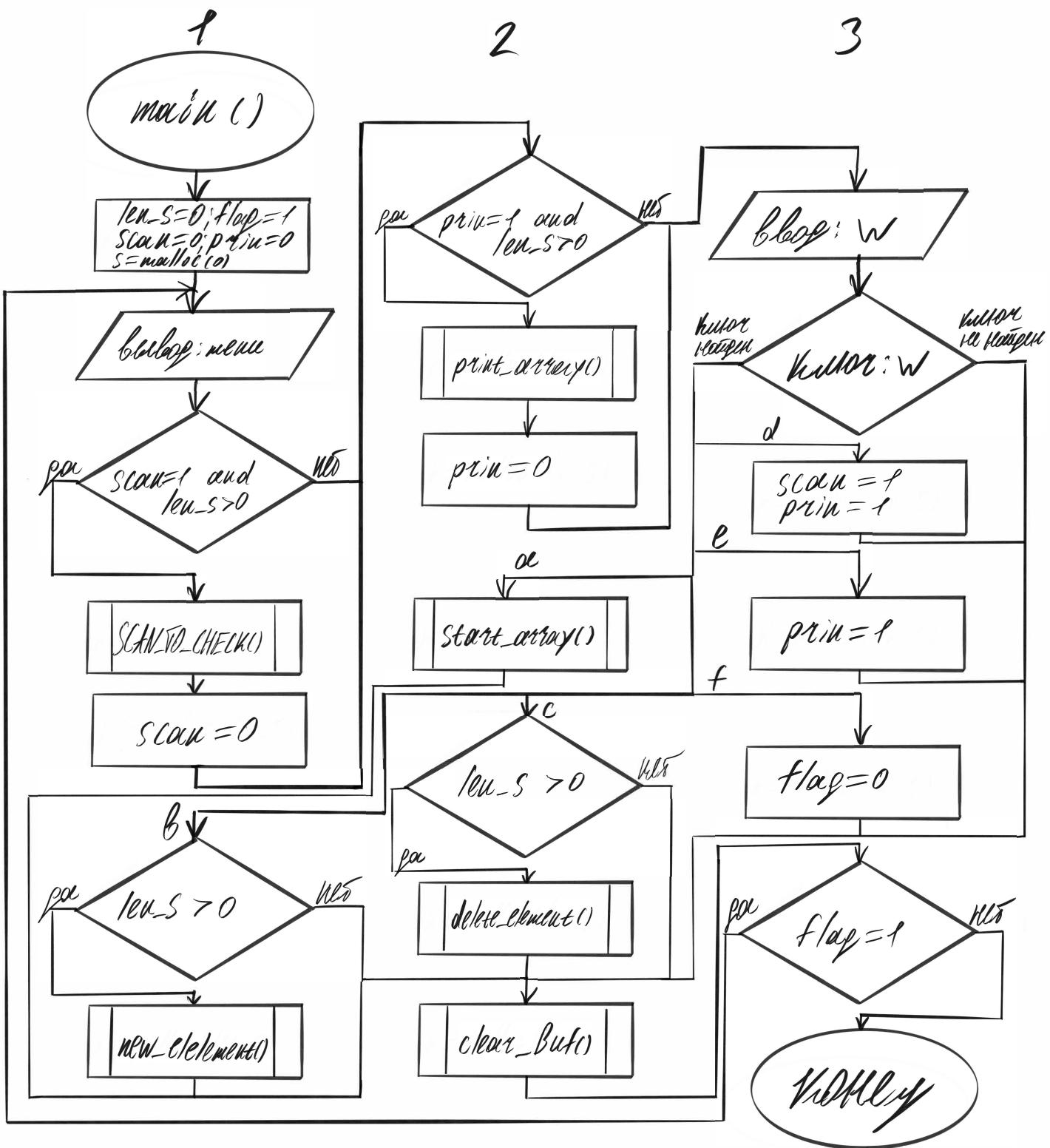
ис. 5: лок-схема функции array.h
(clearBuf)



ис. 6: блок-схема функции
array.h (`new_element`)



ис. 7: блок-схема функции
`array.h (scan_to_check)`



ис. 8: блок-схема функции array.h (main)

4. Исходные коды разработанных программ

Листинг 1: Исходные коды программы Lab3 (файл: lab3.c)

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

Листинг 2: Исходные коды программы Lab3 (файл: array.h)

```
1 #ifndef _ARRAY_
2 #define _ARRAY_
3 void clearBuf(){
4     char w;
5     while ((w=getchar())!='\n' && w !=EOF){}
6 }
7 void start_array(int *s[],int *len_s){
8
9     printf("print len: ");
10    while(scanf("%d",&*len_s)!=1){
11        clearBuf();
12        printf("print len: ");
13    }
14    printf("print elements: ");
15    *s=(int*)realloc(*s,(*len_s)*sizeof(int));
16    for(int i=0;i<*len_s;i++){
17        scanf("%d",&(*s)[i]);
18    }
19
20
21 }
22 void print_array(int *s[],int *len_s){
23     printf("my array: ");
24     for (int i=0;i<*len_s;i++){
25         printf("%d ", (*s)[i]);
26     }
27     printf("\n");
28 }
29 void go_left_to(int *s[],int *len_s, int *i, int *count){
30     if (*count>=2){
31         for (int t=*i;t<*len_s;t++){
32             (*s)[t-(*count)]=(*s)[t];
33         }
34
35         (*len_s)-=(*count);
36         (*i)-=(*count);
37
38         *count=1;
39     }
```

```

41 void scan_to_check(int *s[], int *len_s){
42
43     int *k=(int*)malloc((*len_s)*sizeof(int));
44     int len_k=0, count=1;
45     for (int i=1;i<(*len_s);i++){
46         if ((*s)[i]==(*s)[i-1]){
47             if (count<2){
48                 (k)[len_k]=(*s)[i];
49                 (k)[len_k+1]=(*s)[i-1];
50                 len_k+=2;
51             }else{
52                 (k)[len_k]=(*s)[i];
53                 len_k+=1;
54             }
55             count+=1;
56         }else{
57             go_left_to(&(*s),&(*len_s),&i,&count);
58         }
59     }
60     if (count>=2){*len_s-=count;}
61     *s=(int*)realloc(*s,(*len_s)*sizeof(int));
62     printf("new ");
63     if(len_k>0){print_array(&k,&len_k);}
64     }else{printf("my array empty\n");}
65     free(k);
66 }
67 void print_menu(){
68     printf("\n\n _____ \n");
69     printf(" | liste de ce que je peux faire |\n");
70     printf(" | |\n");
71     printf(" | (a) Инициализация массива |\n");
72     printf(" | (b) Вставка нового элемента по номеру |\n");
73     printf(" | (c) Удаление элемента |\n");
74     printf(" | (d) Индивидуальное задание |\n");
75     printf(" | (e) Вывод содержимого массива |\n");
76     printf(" | Выход из программы Ctrl + D |\n");
77     printf(" | _____ |\n");
78 }

```

```

79 void delete_element(int *s[],int *len_s){
80     int i;
81     do{
82         printf("Enter position: ");
83         scanf("%d",&i);
84     }while(i>*len_s || i==0);
85     for (i;i<*len_s;i++){
86         (*s)[i-1]=(*s)[i];
87     }
88     *len_s-=1;
89     *s=(int*)realloc(*s,(*len_s)*sizeof(int));
90 }
91
92
93 void new_element(int *s[],int *len_s){
94     int i;
95     do{printf("Enter position: ");
96         scanf("%d",&i);
97     }while(i>(*len_s) || i==0);
98     *len_s+=1;
99     *s=(int*)realloc(*s,(*len_s)*sizeof(int));
100    for(int k=(*len_s-1);k>=i;k--){
101        (*s)[k]=(*s)[k-1];
102    }
103    printf("Enter element: ");
104    scanf("%d",&(*s)[i-1]);
105
106
107 }
```

```

108 int main(){
109     int len_s=0, flag=1, scan=0, prin=0;
110     int *s=(int*)malloc(0);
111     char w;
112
113
114     do{
115         system("clear");
116
117         print_menu();
118         if (scan==1){
119             scan_to_check(&s,&len_s);
120             scan=0;
121         }
122         if (len_s>0 && prin==1){
123             print_array(&s,&len_s);
124             prin=0;
125         }
126
127         printf("\nEnter what you want: ");
128         if(scanf("%c",&w)==EOF){w='f';
129         }else if(w=='f'){w='h';}
130
131
132         switch(w){
133             case 'a':
134                 start_array(&s,&len_s); break;
135             case 'b':
136                 if(len_s>0){
137                     new_element(&s,&len_s); }
138                     break;
139             case 'c':
140                 if (len_s>0){
141                     delete_element(&s,&len_s); }
142                     break;
143             case 'd':
144                 scan=1; prin=1; break ;
145             case 'e':
146                 prin=1; break ;
147             case 'f':
148                 flag=0; break;
149             default:
150                 break;
151         }
152         clearBuf();

```

```

153
154
155     }while(flag==1);
156     system("clear");
157
158
159
160
161 }
162 #endif

```

5. Описание тестовых примеров

	а инициализация массива	б вставка нового элемента	с удаление элемента	д индивидуальное задание	е вывод массива
Ввод:	4 1 1 2 4	1 3	2	4 3 1 2 4	4 3 1 2 4
Результат: len; array	4 1 1 2 4	5 3 1 1 2 4	4 3 1 2 4	0	3 1 2 4
Ввод:	6 3 3 3 4 5 6	6 7	4	6 3 3 3 5 7 6	3 5 7 6
Результат: len; array	6 3 3 3 4 5 6	7 3 3 3 4 5 7 6	6 3 3 3 5 7 6	3 3 3	5 7 6
Ввод:	7 5 6 6 7 8 8 5	1 4	8	7 4 5 6 6 7 8 8	3 4 5 7
Результат: len; array	7 5 6 6 7 8 8 5	8 4 5 6 6 7 8 8 5	7 4 5 6 6 7 8 8	4 6 8 8	4 5 7
Ввод:	4 3 3 4 4			4 3 3 4 4	0
Результат: len; array	4 3 3 4 4			4 3 3 4 4	
Ввод:	0			0	0
Результат: len; array				0	
Ввод:	3 3 3 3			3 3 3 3	0
Результат: len; array	3 3 3 3			3 3 3 3	
Ввод:	4 3 4 5 5		1		3 4 5 5
Результат: len; array			3 4 5 5		4 5 5
Ввод:	5 2 3 4 3 2 3 3	5 666			6 2 3 4 3 2 3 666 3
Результат: len; array	5 2 3 4 3 2 3 3	6 2 3 4 3 2 3 666 3			
Ввод:	5 44 55 44 66 66			5 44 55 44 66 66	3 44 55 44
Результат: len; array	5 44 55 44 66 66			2 66 66	44 55 44

6. Скриншоты

```
| liste de ce que je peux faire
```

- | (a) Инициализация массива
- | (b) Вставка нового элемента по номеру
- | (c) Удаление элемента
- | (d) Индивидуальное задание
- | (e) Вывод содержимого массива
- | Выход из программы Ctrl + D

Запуск программы lab3.c

```
Enter what you want: a|
```

```
print len: 4  
print elements: 1 1 2 4|
```

```
Enter what you want: b
```

```
Enter position: 1
```

```
Enter element: 3|
```

```
Enter what you want: c
```

```
Enter position: 2|
```

```
Enter what you want: d|
```

```
new my array empty  
my array: 3 1 2 4|
```

```
my array: 3 1 2 4
```

```
Enter what you want: e|
```

```
| liste de ce que je peux faire
```

- | (a) Инициализация массива
- | (b) Вставка нового элемента по номеру
- | (c) Удаление элемента
- | (d) Индивидуальное задание
- | (e) Вывод содержимого массива
- | Выход из программы Ctrl + D

```
Enter what you want: a
```

```
print len: e  
print len: 4  
print elements: 3 3 4 4|
```

```
Enter what you want: d
```

```
new my array: 3 3 4 4
```

```
-----  
| liste de ce que je peux faire  
|
```

- (a) Инициализация массива
 - (b) Вставка нового элемента по номеру
 - (c) Удаление элемента
 - (d) Индивидуальное задание
 - (e) Вывод содержимого массива
 - Выход из программы Ctrl + D
- ```

```

```
my array: 5 6 6 7 8 8 5
```

```
Enter what you want: a|
```

```
new my array: 6 6 8 8
```

```
my array: 5 7 5
```

```
Enter what you want: d|
```

```

| liste de ce que je peux faire
|
```

- (a) Инициализация массива
  - (b) Вставка нового элемента по номеру
  - (c) Удаление элемента
  - (d) Индивидуальное задание
  - (e) Вывод содержимого массива
  - Выход из программы Ctrl + D
- ```
-----
```

```
Enter what you want: a
```

```
print len: a
```

```
print len: f
```

```
print len: 3
```

```
print elements: 3 3 3
```

```
new my array: 3 3 3
```

```
Enter what you want: d|
```

```
| liste de ce que je peux faire
```

- (a) Инициализация массива
 - (b) Вставка нового элемента по номеру
 - (c) Удаление элемента
 - (d) Индивидуальное задание
 - (e) Вывод содержимого массива
- Выход из программы Ctrl + D

```
Enter what you want: a  
print len: 6  
print elements: 3 3 3 4 5 6
```

```
Enter what you want: b  
Enter position: 6  
Enter element: 7
```

```
my array: 3 3 3 4 5 7 6
```

```
Enter what you want: c  
Enter position: 4
```

```
my array: 3 3 3 5 7 6
```

7. Выводы

В ходе выполнения данной работы на примере программы, выполняющей работу с динамическими массивами были реализованы следующие моменты:

1. Организация ввода массива
2. Удаление элемента из массива
3. Вставка нового элемента
4. Вывод массива
5. Удаление повторяющихся подряд идущий элементов
6. Выход из программы сочетанием: Ctrl+D