

Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №12 «Компьютерные системы и технологии»



# ОТЧЕТ

**О выполнении лабораторной  
работы №4 «Работа со строками»**

**Студент:** Кукса И. В..

**Группа:** Б23-503

**Преподаватель:** Бабалова И. Ф.

Москва — 2023

# 1. Формулировка индивидуального задания

## Индивидуальное задание

К строке, представляющей собой запись математического выражения (операнды — целые числа в десятичной системе счисления, операции — сложение и вычитание), добавить результат его вычисления. Например, строка « $2 + 3 - 10$ » преобразуется в строку « $2 + 3 - 10 = 5$ ».

## 2. Описание использованных типов данных

Из входного потока вводится произвольное количество строк произвольной длины. Каждая строка в общем случае содержит одно или более слов, разделенных пробелами и/или знаками табуляции. Завершение ввода определяется концом файла. Для каждой входной строки формируется новая выходная строка, в которую помещается результат. В полученной строке слова разделяются только одним пробелом, пробелов в её начале и в конце быть не должно. Введённая и сформированная строки выводятся на экран в двойных кавычках.

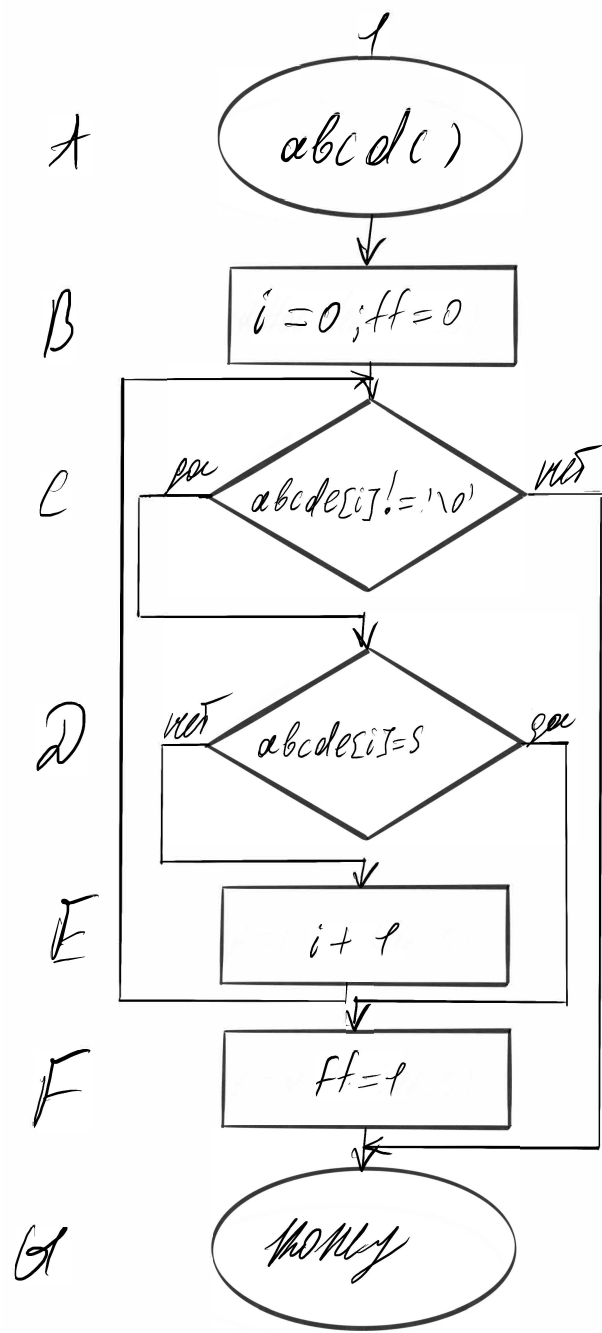
В ходе выполнения лабораторной работы должны быть разработаны:

1. Программа, использующая функцию `getline()` из состава библиотеки GNU readline для ввода строк и функции стандартной библиотеки для их обработки (`<string.h>`).
2. Программа, идентичная п. 1, за исключением того, что все библиотечные функции заменены на собственную реализацию данных функций, представленную в отдельных файлах (например: `mystring.h`, `mystring.c`).

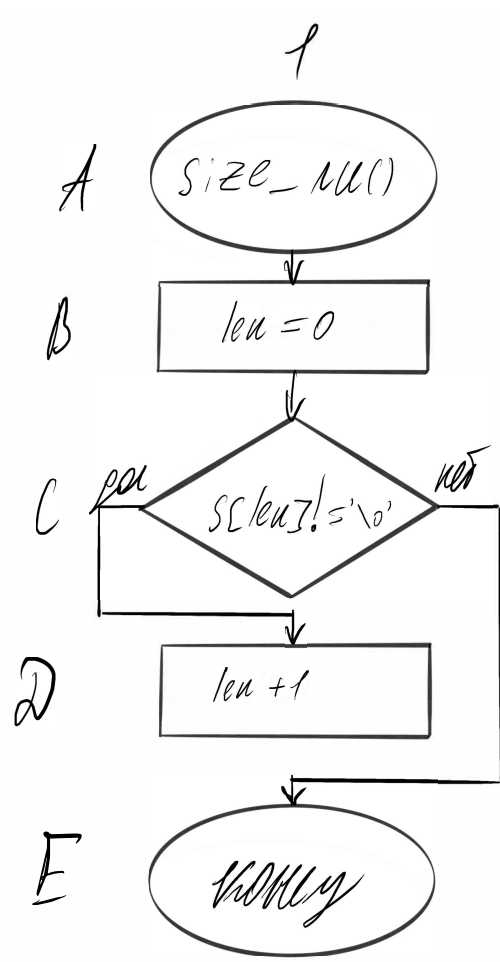
Отчётность по выполнению лабораторной работы должна включать:

1. Блок-схему алгоритма работы основной программы.
2. Блок-схемы алгоритмов работы функций по обработке строк.
3. Исходные коды всех программ.
4. Тестовые наборы для программ п. 1 и п. 2.
5. Сравнительный анализ времени, потраченного на решение задачи программами п. 1 и п. 2 (на конкретных примерах).

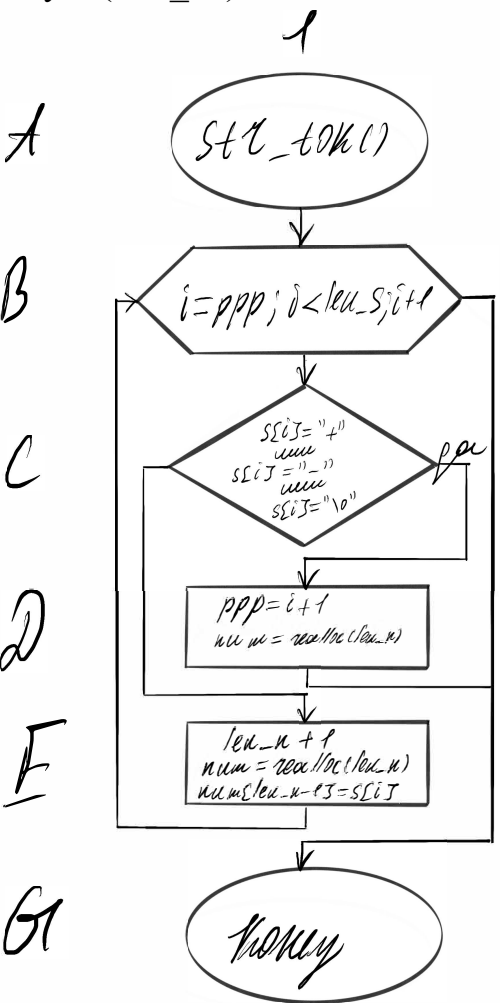
3. Описание использованного алгоритма



Ррис. 2: Блок-схема функции array.h (abcd)



Ррис. 1: Блок-схема функции array.h (size\_nu)



Ррис. 3: Блок-схема функции array.h (str\_tok)

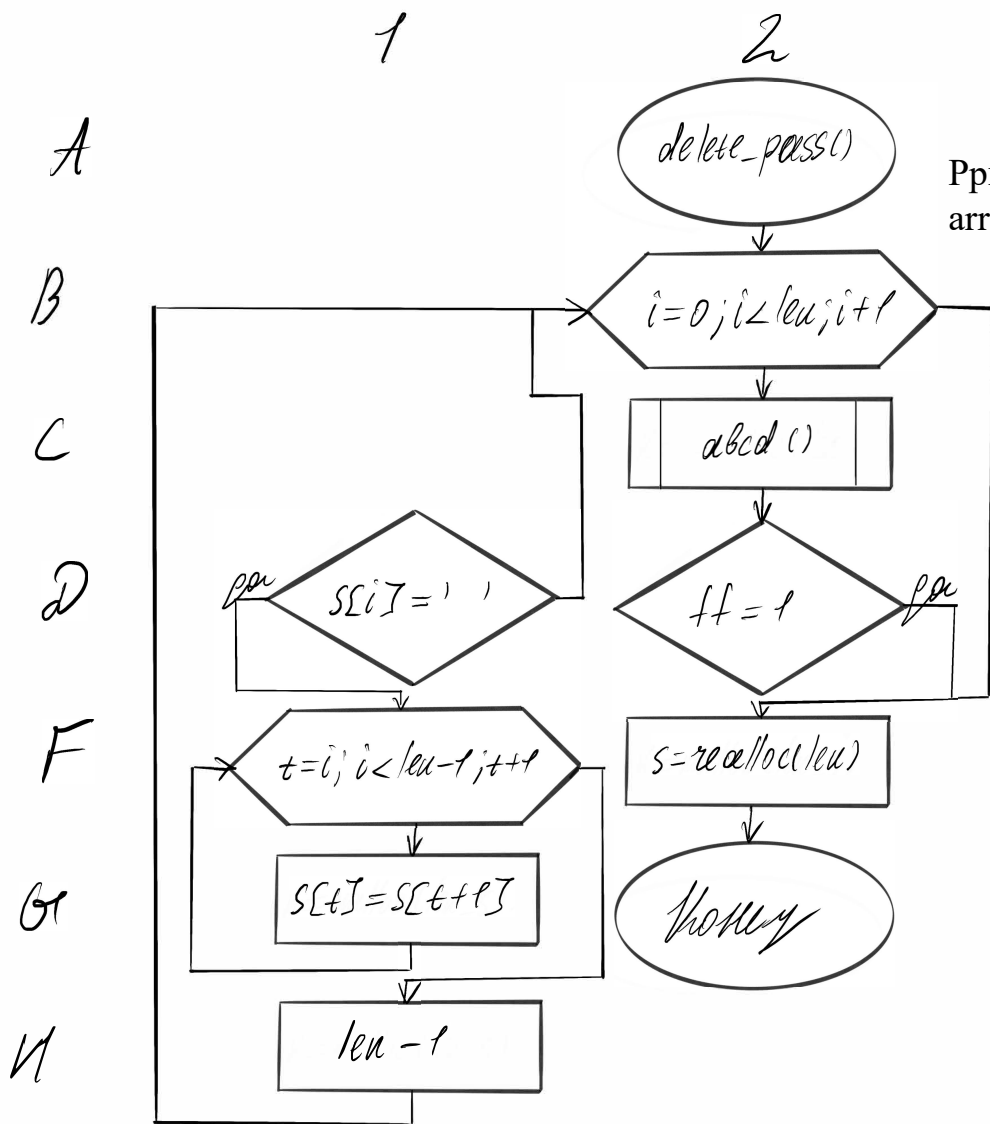
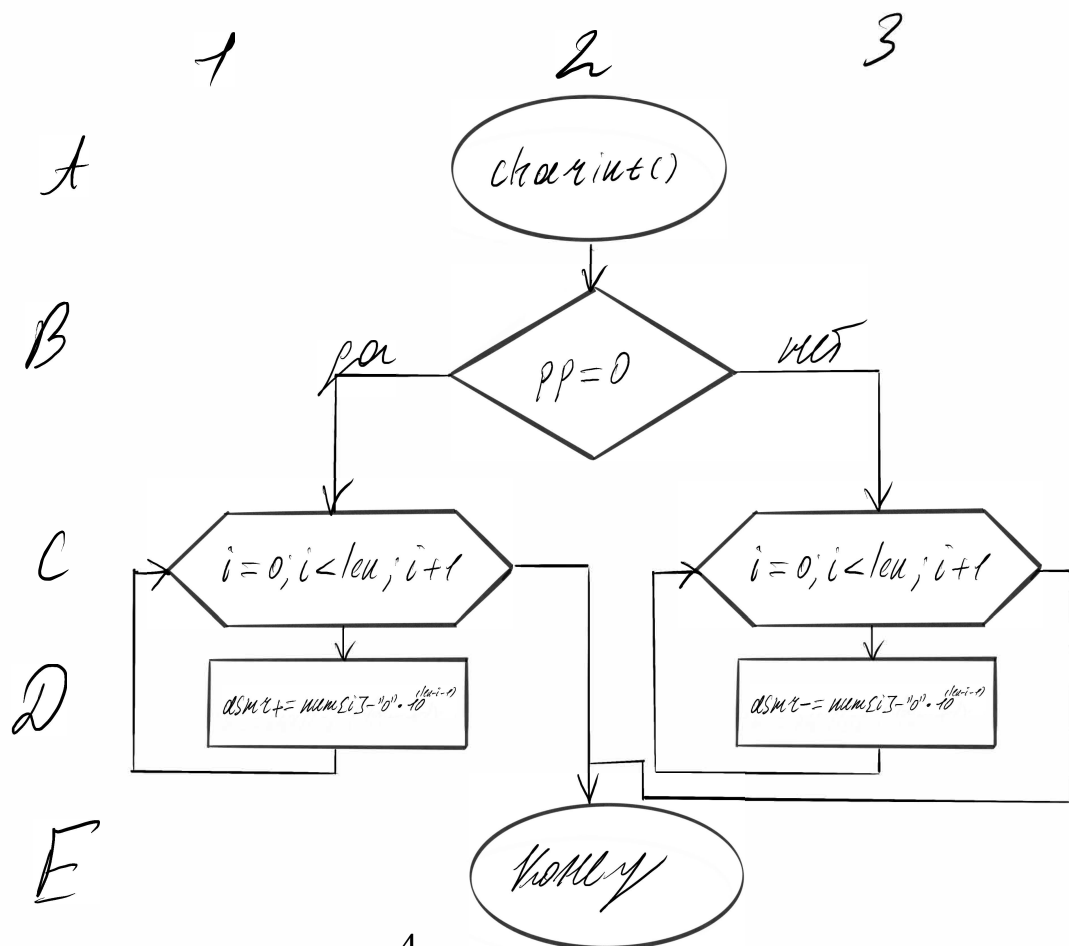


Рис. 5: Блок-схема функции array.h (charint)



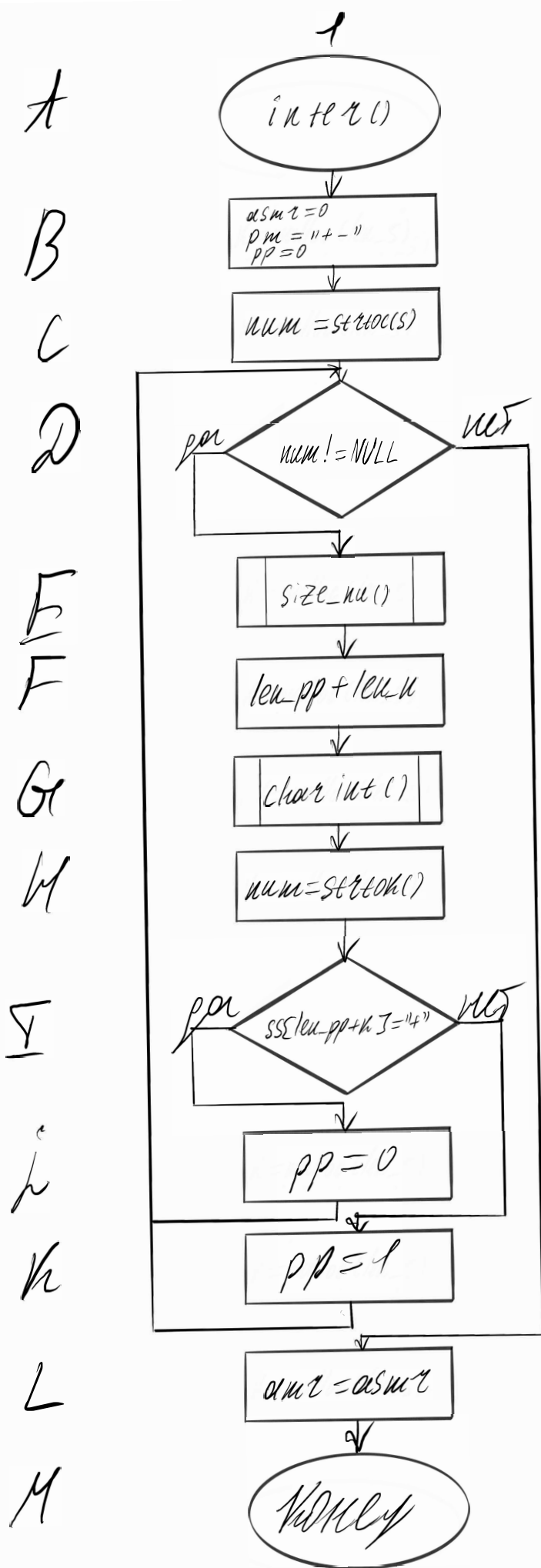


Рис. 7: Блок-схема функции array.h (inter)

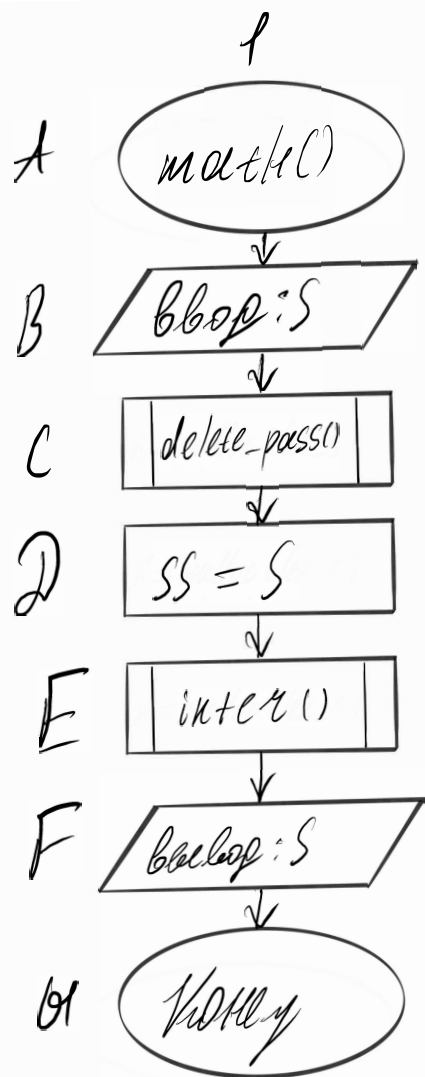
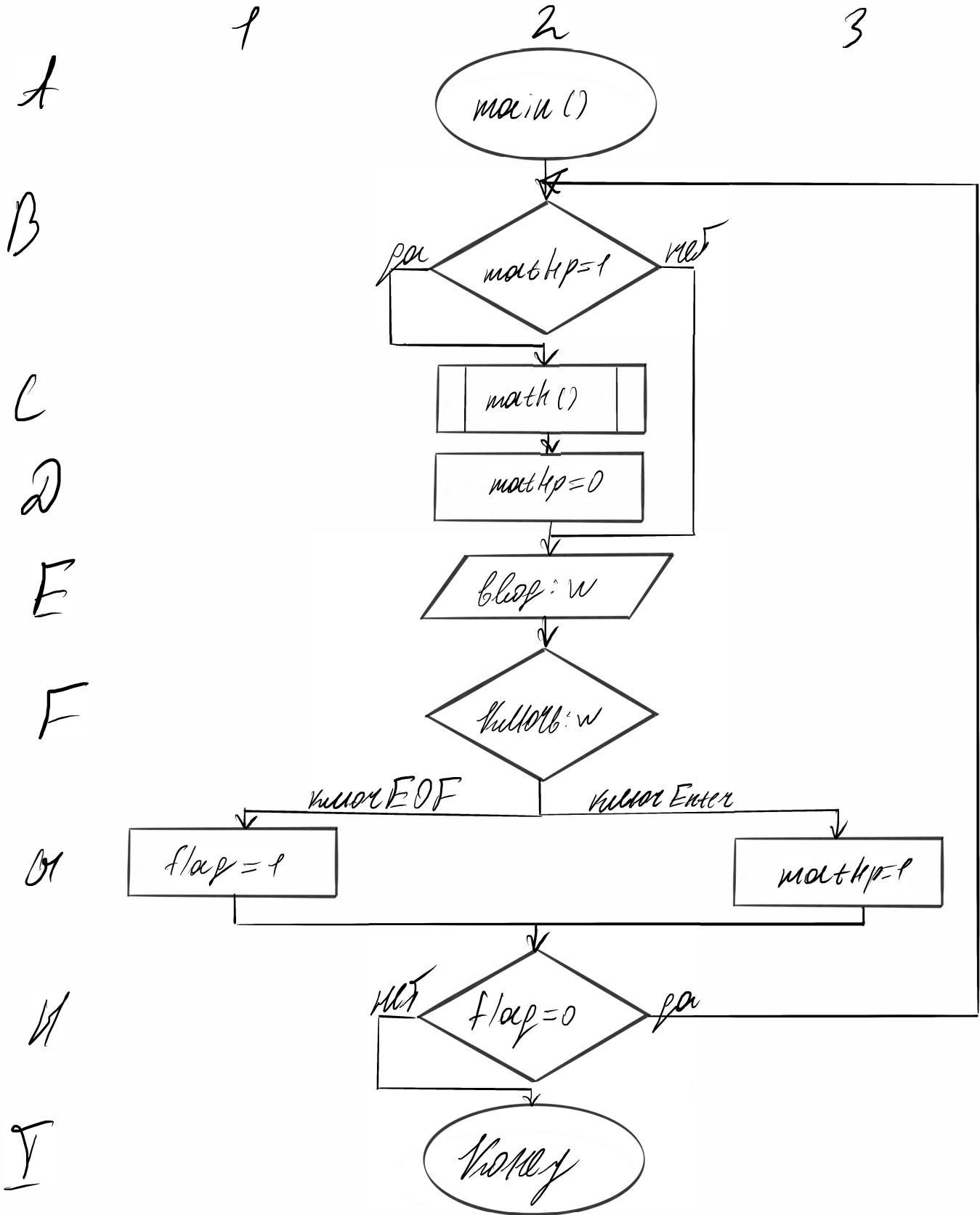


Рис. 6: Блок-схема функции array.h (math)



Ррис. 8: Блок-схема  
функции main.c (main)

## 4. Исходные коды разработанных программ

Листинг 1: Исходные коды программы Lab4\_1 (файл: lab4.c)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <math.h>
5  #include <time.h>
6  #include "array.h"
7
```

Листинг 2: Исходные коды программы Lab4\_1 (файл: array.h)

```
1  #ifndef _LIB_ARRAY_
2  #define _LIB_ARRAY_
3
4  void print_menu();
5  void size_nu(char *s[],int *len);
6  void delete_pass(char *s[],int *len);
7  void charint(char *num[],int *len,int *asmr,int *pp);
8  void inter(char *s[],int *len,char *ss[],int *amr);
9  void math();
10 #endif
```

Листинг 3: Исходные коды программы Lab4\_1 (файл: array.c)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <math.h>
5  #include <time.h>
6  #include "array.h"
7
8  void print_menu(){
9      printf("                \n");
10     printf("| Привет, я умнею решать выражения используя | \n");
11     printf("| операции сложения и вычитания           | \n");
12     printf("|                | \n");
13     printf("| -Enter чтобы ввести выражение             | \n");
14     printf("| -Ctrl + D чтобы завершить программу       | \n");
15     printf("|                | \n\n");
16 }
17
18
19 void size_nu(char *s[],int *len){
20     (*len)=0;
21
22     while((*s)[*len]!='\0'){
23         (*len)++;
24     }
25 }
26 void abcd(char s,char abcde[],int *ff){
27     int i=0;
28     *ff=0;
29     while(abcde[i]!='\0'){
30         if(abcde[i]==s){
31             *ff=1;
32             break;
33         }
34
35         i++;
36     }
37 }
38 }
```

```

39 void delete_pass(char *s[],int *len,int *ff){
40     char abcde[]="qwertyuiopadfgjhklzxcvbnмйцукенгшзхъёфывапрлджэячсмитьбю/|\0";
41     for (int i=0;i<*len;i++){
42         abcd((*s)[i],abcde,&(*ff));
43         if((*ff)==1){
44             break;
45         }
46         if ((*s)[i]==' '){
47             for (int t=i;t<(*len)-1;t++){
48                 (*s)[t]=(*s)[t+1];
49             }
50             *len-=1;
51         }
52         if (i<*len-1 && (((*s)[i]=='-' && (*s)[i+1]=='-') || ((*s)[i]=='+' && (*s)[i+1]=='+'))){
53             for (int t=i;t<(*len)-1;t++){
54                 (*s)[t]=(*s)[t+1];
55             }
56             *len-=1;
57         }
58     }
59     (*s)=(char*)realloc(*s,*len);
60 }
61
62 void charint(char *num[],int *len,int *asmr,int *pp){
63     if((*pp)==0){
64         for (int i=0;i<*len;i++){
65             (*asmr)+=(((*num)[i]-'0')*pow(10,(*len)-i-1));
66         }
67     }else{
68         for (int i=0;i<*len;i++){
69             (*asmr)-=(((*num)[i]-'0')*pow(10,(*len)-i-1));
70         }
71     }
72 }
73 void inter(char *s[],int *len,char *ss[],int *amr){
74     int asmr=0;
75     char pm[]="+-";
76     int len_n,len_pp=0,pp=0,k=0;
77
78     char *num=strtok((*s),pm);
79
80     while(num!=NULL){
81         size_nu(&num,&len_n);
82         len_pp+=len_n;
83
84         charint(&num,&len_n,&asmr,&pp);
85
86         num=strtok(NULL,pm);
87
88         if((*ss)[len_pp+k]=='+'){
89             pp=0;
90         }else{
91             pp=1;
92         }
93         k++;
94     }
95     (*amr)=asmr;
96 }

```



```

98 void math(){
99     char *s = NULL, c,*ss=NULL;
100     int len ,asmr,ff=0;
101     ss = (char*) malloc(sizeof(char));
102     clock_t start, fin;
103     double time;
104
105     len=1;
106     s = (char*) malloc(sizeof(char));
107     printf("Введите выражение: ");
108
109
110
111     while((c=getchar()) != '\n') {
112         s[len - 1] = c;
113         len++;
114
115         s= (char*) realloc(s, len);
116     }
117     s[len - 1] = '\0';
118     fflush(stdin);
119     start=clock();
120     delete_pass(&s,&len,&ff);
121
122
123     while(ff==1){
124         system("cls");
125         print_menu();
126
127         len=1;
128         s = (char*) malloc(sizeof(char));
129         printf("Есть нечитаемые символы\nВведите выражение: ");
130
131
132
133         while((c=getchar()) != '\n') {
134             s[len - 1] = c;
135             len++;
136
137             s= (char*) realloc(s, len);
138         }
139         s[len - 1] = '\0';
140         fflush(stdin);
141
142         delete_pass(&s,&len,&ff);
143     }
144
145     memcpy(ss,s,len);
146     inter(&s,&len,&ss,&asmr);
147     system("cls");
148     print_menu();
149     printf("ОТВЕТ: %s=%d\n", ss, asmr);
150     fin=clock();
151     time=(double)(fin-start);
152     printf("%f\n",time);
153     free(s);
154     free(ss);
155 }

```

## Листинг 4: Исходные коды программы Lab4\_1 (файл: main.c)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include "array.h"
4
5  int main(){
6      char w;
7      int flag=0,mathp=0;
8
9
10
11     do{
12         system("cls");
13         print_menu();
14         if(mathp==1){
15             math();
16             mathp=0;
17         }
18
19
20
21
22
23     printf("Что мне делать: ");
24     if(scanf("%c",&w)==EOF){w='f';
25     }else if (w=='f'){
26         w='p';
27     }
28     fflush(stdin);
29     while(w!='f' && w!='\n'){
30         system("cls");
31         print_menu();
32         printf("я не могу распознать такое\n");
33         printf("Введите один из вариантов: ");
34         if(scanf("%c",&w)==EOF){w='f';
35         }else if (w=='f'){
36             w='p';
37         }
38         fflush(stdin);
39     }
40     switch (w)
41     {
42         case 'f':
43             flag=1; break;
44
45         default:
46             mathp=1; break;
47     }
48
49     }while(flag==0);
50
51     return 0;
52 }
```

## 4. Исходные коды разработанных программ

### Листинг 1: Исходные коды программы Lab4\_2 (файл: lab42.c)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <math.h>
5  #include <time.h>
6  #include "array.h"
```

### Листинг 2: Исходные коды программы Lab4\_2 (файл: array2.h)

```
1  #ifndef _LIB_ARRAY_
2  #define LIB_ARRAY
3  void str_tok(char *num[],char *s[],int *len_s,int *len_n,char pm[],int *ppp);
4  void print_menu();
5  void size_nu(char *s[],int *len);
6  void delete_pass(char *s[],int *len);
7  void charint(char *num[],int *len,int *asmr,int *pp);
8  void inter(char *s[],int *len,char *ss[],int *amr);
9  void math();
10 #endif
```

### Листинг 3: Исходные коды программы Lab4\_2 (файл: array2.c)

```
7  void str_tok(char *num[],char *s[],int *len_s,int *len_n,char pm[],int *ppp){
8
9      (*len_n)=0;
10     (*num)=NULL;
11
12     for (int i=(*ppp);i<*len_s;i++){
13         if ((*s)[i]==(pm)[0] || (*s)[i]==(pm)[1] || (*s)[i]=='\0'){
14             *ppp=i+1;
15             (*num)=(char*)realloc(*num,(*len_n+1)*sizeof(char));
16             (*num)[*len_n]='\0';
17             break;
18         }
19         (*len_n)+=1;
20         (*num)=(char*)realloc(*num,(*len_n)*sizeof(char));
21         (*num)[*len_n-1]=(*s)[i];
22     }
23 }
24
25 }
26
27
28
29 void print_menu(){
30     printf("\n");
31     printf("Привет, я умнею решать выражения используя \n");
32     printf("операции сложения и вычитания \n");
33     printf("\n");
34     printf("-Enter чтобы ввести выражение \n");
35     printf("-Ctrl + D чтобы завершить программу \n");
36     printf("\n\n");
37 }
38
39
40 void size_nu(char *s[],int *len){
41     (*len)=0;
42
43     while((*s)[*len]!='\0'){
44         (*len)++;
45     }
46 }
47 void abcd(char s,char abcde[],int *ff){
48     int i=0;
49     *ff=0;
50     while(abcde[i]!='\0'){
51         if(abcde[i]==s){
52             *ff=1;
53             break;
54         }
55
56         i++;
57     }
58 }
59
60 }
```

```

39 void delete_pass(char *s[],int *len,int *ff){
40     char abcde[]="qwertyuiopadfgjhklzxcvbnмйцукенгшзхъёфывапрлджячсмитьбю/|\0";
41     for (int i=0;i<*len;i++){
42         abcd((*s)[i],abcde,&(*ff));
43         if((*ff)==1){
44             break;
45         }
46         if ((*s)[i]==' '){
47             for (int t=i;t<(*len)-1;t++){
48                 (*s)[t]=(*s)[t+1];
49             }
50             *len-=1;
51         }
52         if (i<*len-1 && (((*s)[i]=='-' && (*s)[i+1]=='-') || ((*s)[i]=='+' && (*s)[i+1]=='+'))){
53             for (int t=i;t<(*len)-1;t++){
54                 (*s)[t]=(*s)[t+1];
55             }
56             *len-=1;
57         }
58     }
59     (*s)=(char*)realloc(*s,*len);
60 }
61
62 void charint(char *num[],int *len,int *asmr,int *pp){
63     if((*pp)==0){
64         for (int i=0;i<*len;i++){
65             (*asmr)+=(((*num)[i]-'0')*pow(10,(*len)-i-1));
66         }
67     }else{
68         for (int i=0;i<*len;i++){
69             (*asmr)-=(((*num)[i]-'0')*pow(10,(*len)-i-1));
70         }
71     }
72 }
73 void inter(char *s[],int *len,char *ss[],int *amr){
74     int asmr=0;
75     char pm[]="+-";
76     int len_n,len_pp=0,pp=0,k=0;
77
78     char *num=strtok((*s),pm);
79
80     while(num!=NULL){
81         size_nu(&num,&len_n);
82         len_pp+=len_n;
83
84         charint(&num,&len_n,&asmr,&pp);
85
86         num=strtok(NULL,pm);
87
88         if((*ss)[len_pp+k]=='+'){
89             pp=0;
90         }else{
91             pp=1;
92         }
93         k++;
94     }
95     (*amr)=asmr;
96 }

```

```

98 void math(){
99     char *s = NULL, c,*ss=NULL;
100     int len ,asmr,ff=0;
101     ss = (char*) malloc(sizeof(char));
102     clock_t start, fin;
103     double time;
104
105     len=1;
106     s = (char*) malloc(sizeof(char));
107     printf("Введите выражение: ");
108
109
110
111     while((c=getchar()) != '\n') {
112         s[len - 1] = c;
113         len++;
114
115         s= (char*) realloc(s, len);
116     }
117     s[len - 1] = '\0';
118     fflush(stdin);
119     start=clock();
120     delete_pass(&s,&len,&ff);
121
122
123     while(ff==1){
124         system("cls");
125         print_menu();
126
127         len=1;
128         s = (char*) malloc(sizeof(char));
129         printf("Есть нечитаемые символы\nВведите выражение: ");
130
131
132
133         while((c=getchar()) != '\n') {
134             s[len - 1] = c;
135             len++;
136
137             s= (char*) realloc(s, len);
138         }
139         s[len - 1] = '\0';
140         fflush(stdin);
141
142         delete_pass(&s,&len,&ff);
143     }
144
145     memcpy(ss,s,len);
146     inter(&s,&len,&ss,&asmr);
147     system("cls");
148     print_menu();
149     printf("ОТВЕТ: %s=%d\n", ss, asmr);
150     fin=clock();
151     time=(double)(fin-start);
152     printf("%f\n",time);
153     free(s);
154     free(ss);
155 }

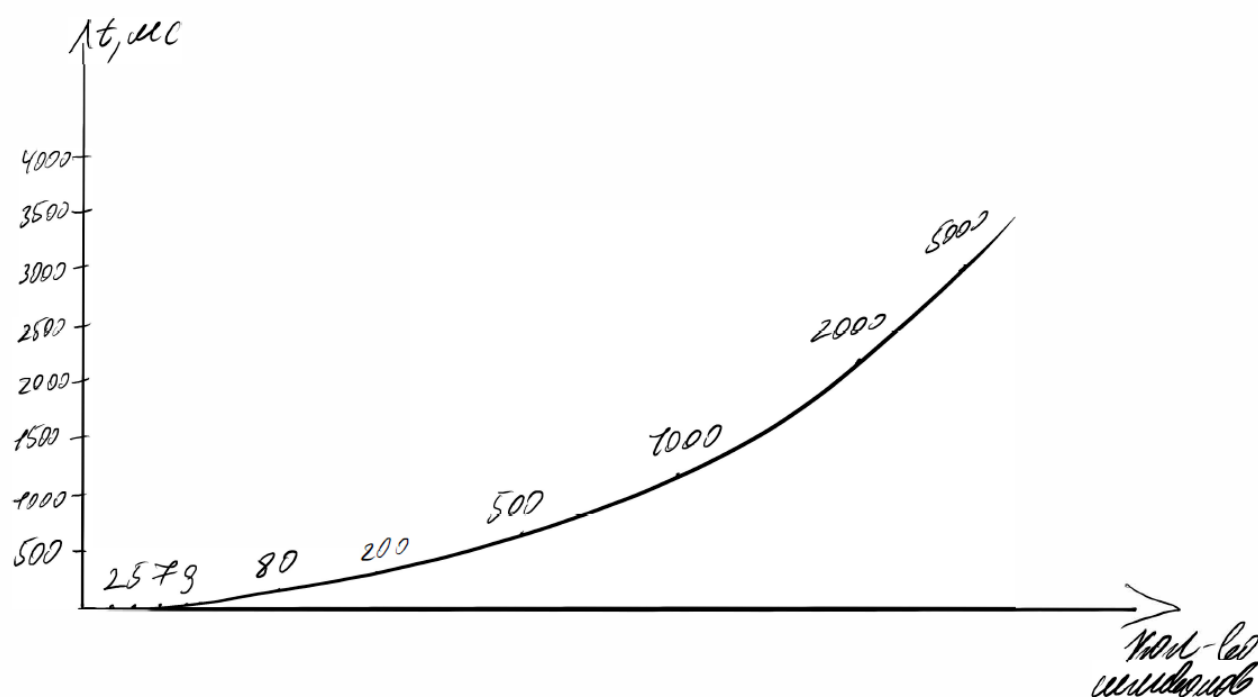
```

## Листинг 4: Исходные коды программы Lab4\_2 (файл: main.c)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include "array.h"
4
5  int main(){
6      char w;
7      int flag=0,mathp=0;
8
9
10
11     do{
12         system("cls");
13         print_menu();
14         if(mathp==1){
15             math();
16             mathp=0;
17         }
18
19
20
21
22
23     printf("Что мне делать: ");
24     if(scanf("%c",&w)==EOF){w='f';
25     }else if (w=='f'){
26         w='p';
27     }
28     fflush(stdin);
29     while(w!='f' && w!='\n'){
30         system("cls");
31         print_menu();
32         printf("я не могу распознать такое\n");
33         printf("Введите один из вариантов: ");
34         if(scanf("%c",&w)==EOF){w='f';
35         }else if (w=='f'){
36             w='p';
37         }
38         fflush(stdin);
39     }
40     switch (w)
41     {
42         case 'f':
43             flag=1; break;
44
45         default:
46             mathp=1; break;
47     }
48
49     }while(flag==0);
50
51     return 0;
52 }
```

## 5. Описание тестовых примеров

Входная строка	Выходная строка
9+8	9+8=17
9999999-4-4-44-4	9999999-4-4-44-4=9999943
44-4-4-4-4-4-4-4-4-4-4-4-4-4	44-4-4-4-4-4-4-4-4-4-4-4-4-4=-8
5555555-555555+33	5555555-555555+33=33
12345678-0987654456	12345678-0987654456=-975308778
8765856-7564747645547	8765856-7564747645547=-2147483648
657575799-4567654	657575799-4567654=653008145
-9=-9	-9=-9
-9-9-9-9-9-9-9-9-9-9-9-9-9=-126	-9-9-9-9-9-9-9-9-9-9-9-9-9=-126



## 6. Скриншоты

Запуск и сборка программы  
lab4

```
Привет, я умею решать выражения используя  
операции сложения и вычитания
```

```
-Enter чтобы ввести выражение  
-Ctrl + D чтобы завершить программу
```

Ответ: 9+8=17

Ответ: 9999999-4-4-44-4=9999943

Ответ: 44-4-4-4-4-4-4-4-4-4-4=-8

Ответ: 5555555-5555555+33=33

Ответ: 12345678-0987654456=-975308778

Ответ: -9=-9

## 7. Выводы

В ходе выполнения данной работы на примере программы, выполняющей работу со строками были реализованы следующие моменты:

1. Организация ввода строки без `getline()`
2. Перевод `char` в `int`
3. Был самостоятельно реализован `strtok()`
4. Вывод строки и результата выражения
5. Выход из программы сочетанием: `Ctrl+D`