

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра Вычислительной техники

ОТЧЕТ
по лабораторной работе № 5
по дисциплине «Программирование»
Тема: Обработка строк

Студент гр. 3311

Баймухамедов Р.Р.

Преподаватель

Хахаев И. А.

Санкт-Петербург

2023

Цель работы.

Целью работы является получение практических навыков в обработке строк на языке программирования C.

Задание (вариант 4)

Ввести строку текста, состоящую из слов и символов-разделителей, находящихся в произвольном количестве до и после слов, а затем строку символов-разделителей между словами. Подсчитать и вывести количество символов в предпоследнем слове.

Постановка задачи и описание решения

Для выполнения данной лабораторной работы необходимо разработать программу, которая будет сначала записывать строку, а затем анализировать её. Для достижения этой цели необходимо выполнить следующие шаги:

1. Запросить у пользователя значение строки текста, состоящую из слов и символов-разделителей, находящихся в произвольном кол-ве до и после слов.
2. Используя функцию `new_gets (string, MAXLEN)`, сохраняем полученное значение в `ostring` строку длиной `MAXLEN = 80`.
3. Запросить у пользователя значение строки символов-разделителей
4. Используя функцию `new_gets (string, MAXLEN)`, сохраняем полученное значение в `sep` строку длиной `MAXLEN = 80`.
5. Посимвольно проверять строку `ostring`, начиная с конца, на нахождение символов-разделителей, если это удачно, то увеличиваем значение `flag` на единицу и `nword=1`.
6. Если при посимвольной проверке строки `ostring` значение элемента находится в промежутках кодов-символов, являющихся буквами в ASCII, и `flag=1` (т.е. ранее встречалась одна последовательность символов-разделителей), то символы, встречаемые до следующей

последовательности символов-разделителей, являются предпоследним словом.

7. Подсчитываем количество символов, встречаемых до следующей последовательности символов-разделителей и выводим это значение.

Описание переменных

Функция void new_gets (char s[], int lim)

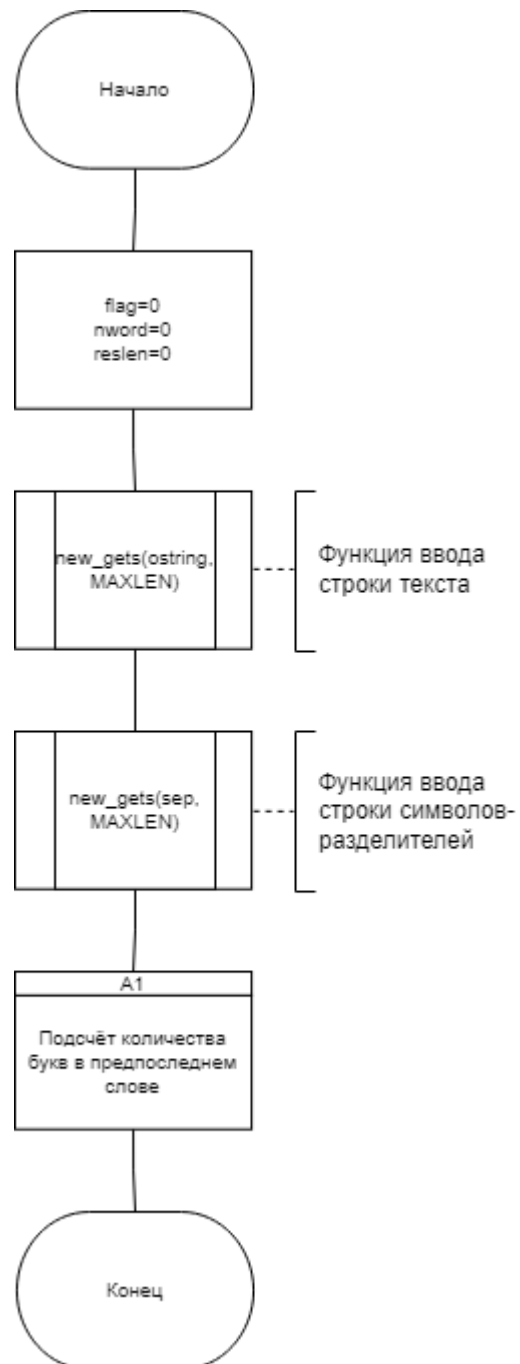
№	Имя переменной	Тип	Назначение
1	c	char	Вводимый символ
2	i	int	Переменная в цикле, отвечающая за индекс вводимого символа в строке

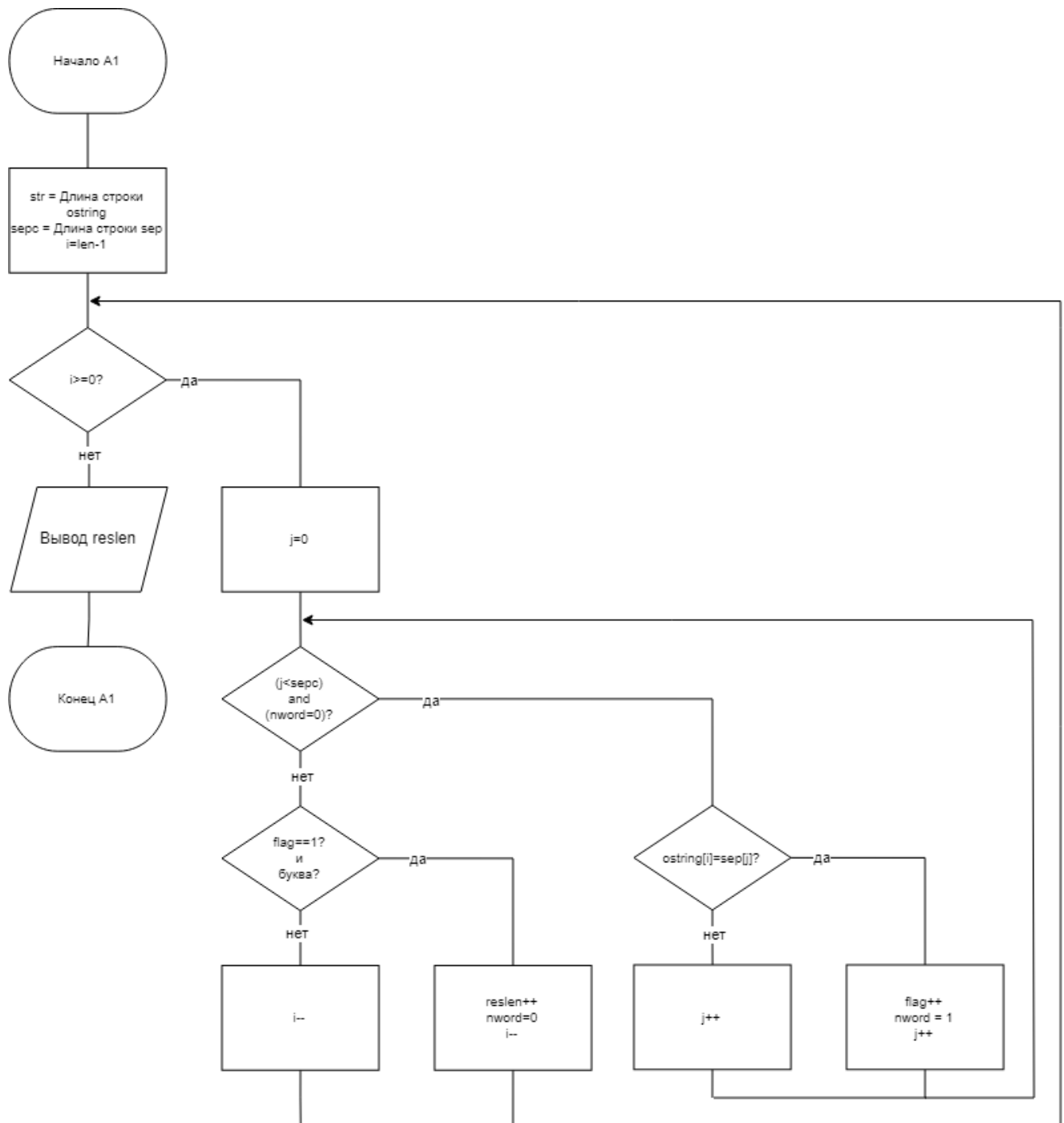
main

№	Имя переменной	Тип	Назначение
1	ostring[MAXLEN]	char	Исходная строка
2	sep[MAXLEN]	char	Строка символов-разделителей
3	len	int	Длина исходной строки
4	sepc	int	Длина строки символов-разделителей
5	flag	int	Индикатор разделения между словами
6	nword	int	Индикатор последовательности, показывающий, что она состоит из символов-разделителей
7	i	int	Вспомогательная переменная, отвечающая за символ исходной строки
8	j	int	Вспомогательная переменная, отвечающая за символ строки-разделителей

9	reslen	int	Количество символов в предпоследнем слове
---	--------	-----	---

Схема алгоритма





Контрольные примеры

Пример 1:

Исходные данные:

Original string: “apple banana cat”

String of separators: “ “

Результаты:

Number of letter in penultimate word: 6

Пример 2:

Исходные данные:

Original string: "pineapple,mango,dog"

String of separators: ",",

Результаты:

Number of letter in penultimate word: 5

Пример 3:

Исходные данные:

Original string: "potato,. .,meat,,,... .chicken"

String of separators: ",."

Результаты:

Number of letter in penultimate word: 4

Текст программы

```
#include <stdio.h>
#include <string.h>
#define MAXLEN 80
/*Function of input string*/
void new_gets(char s[], int lim)
{
    char c;
    int i=0;
    while((((c=getchar())!='\n')&&(i<lim-1))
    {
        s[i]=c;
        i++;
    }
}
```

```

        s[i]='\0';
    }

/*Main program*/
int main(){
    char ostring[MAXLEN], sep[MAXLEN];
    int len, sepc, flag=0, nword=0, i, j, reslen=0;

    /*Input original string and separator string*/
    printf("Enter the string:\n");
    new_gets(ostring,MAXLEN);
    printf("\nEnter the string of separators:\n");
    new_gets(sep,MAXLEN);

    /*Counting*/
    len=strlen(ostring);
    sepc=strlen(sep);
    for(i=len-1;i>=0;i--){
        for(j=0;(j<sepc)&&(nword==0);j++){
            if(ostring[i]==sep[j]){
                flag=flag+1;
                nword=1;
            }
        }
        if ((flag == 1)&&((ostring[i] >= 65 && ostring[i] <= 90) || (ostring[i] >=
97 && ostring[i] <= 122)))){
            reslen++;
            nword=0;
        }
    }
}

```

```
/*Output*/  
printf("\nNumber of letters in penultimate word: %d", reslen);  
return 0;  
}
```

Примеры выполнения программы

```
Enter the string:  
apple banana cat  
  
Enter the string of separators:  
  
Number of letters in penultimate word: 6
```

```
Enter the string:  
pineapple,mango,dog  
  
Enter the string of separators:  
,  
  
Number of letters in penultimate word: 5
```

```
Enter the string:  
potato,. .,meat,,,... .chicken  
  
Enter the string of separators:  
, .  
  
Number of letters in penultimate word: 4
```

Вывод

В результате выполнения работы изучены строки на языке программирования С и получены практические навыки в их применении.