

Министерство образования Республики Беларусь

Учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»

Кафедра электронных вычислительных машин

Отчёт  
по лабораторной работе № 6  
«Символьные строки. Массивы символьных строк»  
по курсу:  
«Основы алгоритмизации и программирования»

Выполнил:  
Студент группы 950503  
Полховский А.Ф.

Проверил:  
Дубовский А.Л.

Минск 2019

Цель работы: Освоить методику работы с символьными строками. Научиться использовать функции для работы со строками. Научиться объявлять и использовать массивы строк и функции для работы со строками.

Вариант 7

Условие:

8) Даны натуральное число  $n$  и символы  $s_1, \dots, s_n$ . Удалить из каждой группы идущих подряд цифр, в которой более двух цифр и которой предшествует точка, все цифры, начиная с третьей (например,  $ab+0.1973-1.1$  преобразуется в  $ab+0.19-1.1$ ).

9) Дан текст. Выделить все слова в предложениях, состоящих только из цифр. Определить сумму чисел во всех предложениях.

Код программ на языке Си:

8)

```
#include <stdio.h>
#include <conio.h>
#include <locale.h>

int main()
{
    setlocale(LC_ALL, "RUSSIAN");
    int i = 0, n, k = 0, c;
    puts("Введите размерность строки n ");
    while (!scanf_s("%d", &n)) rewind(stdin);
    n++; // для 0 символа
    char* string = (char*)malloc(n*sizeof(char));
    puts("\nВводите строку ");
    rewind(stdin);
    while (--n > 0 && (c = getchar()) != EOF && c != '\n') string[i++] = c;
    string[i] = '\0';
    printf("\nИсходная строка:\n%s\n", string);

    int j = 0, numb;
    i=0;
    while (string[i++] != '\0')
        if (string[i] == '.')
            if (string[i+1] >= 48 && string[i+1] <= 57)
            {
                j = i+1; // i - индекс т. , j - индекс цифры после т.
                numb = 0; // счетчик
                while (string[j] >= '0' && string[j] <= '9')
                {
                    j++; //индекс последней цифры
                    numb++; // количество цифр после т.
                }
                if (j >= i + 2) // если цифр больше чем 2
                {
                    k = j - numb + 2; //начиная с этого элемента удалить
                    //все остальные
                    while (string[k] != '\0')
                    {
                        string[k] = string[k + numb - 2];
                        k++;
                    }
                }
            }
    printf("\nПреобразование строка: %s\n", string);
```

```

        system("PAUSE");
        return 0;
    }

9)
#include <conio.h>
#include <stdio.h>
#include <string.h>
#include <locale.h>
#include <stdlib.h>
//Дан текст.Выделить все слова в предложениях, состоящих только из цифр.
//Определить сумму чисел во всех предложениях.

int Atoi(char *nums) {
    setlocale(LC_ALL, "RUSSIAN");
    int num = 0, i = 0;
    while (nums[i] != '\0')
    {
        num = num * 10 + nums[i] - '0';
        i++;
    }
    return num;
}

int main()
{
    setlocale(LC_ALL, "RUSSIAN");
    char c;
    int Size = 100;
    int j = 0, i = 0, len = 0, k = 0;
    char* string = (char*)malloc(Size * sizeof(char));
    puts("\nВводите текст ");
    rewind(stdin);
    do {
        c = getchar();
        string[i++] = c;
        if (i + 1 == Size) {
            Size *= 2;
            string = (char*)realloc(Size, sizeof(char));
        }
    } while (c != '\n');
    string[i - 1] = '\0';
    printf("\nИсходная строка:\n%s", string);

    i = 0;
    int sum = 0;
    char* num = (char*)calloc(10, sizeof(char));
    int numberLength = 0;
    while (string[i] != '\0') {
        if (string[i] >= '0' && string[i] <= '9') { // запись числа
            num[numberLength] = string[i];
            numberLength++;
        }
        if (i != 0)
            if ((string[i - 1] < '0' || string[i - 1] > '9') && string[i - 1] !=
' ' && string[i - 1] != '.') {
                numberLength = 0; // удаляем число (f10 - не число)
                while (string[i] >= '0' && string[i] <= '9')
                    i++;
            }
        if (string[i] == ' ' || string[i] == '.' || string[i + 1] == '\0') // i+1
для while(не работает посл раз)
            if (numberLength) {
                num[numberLength] = '\0';

```

```
        numberLength = 0;
        printf("\nчисло %d", Atoi(num));
        sum += Atoi(num);
    }
    i++;
}

printf("\nСумма чисел %d", sum);
system("PAUSE");
return 0;
```

Вывод: изучил краткие теоретические сведения по данной теме; по ним разработал, написал и отладил работу программ 7-ого варианта.