

ГУАП

КАФЕДРА № 14

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

доц., канд. техн. наук  
\_\_\_\_\_  
должность, уч. степень, звание

/\_\_\_\_\_  
подпись, дата

Е.П. Овсянников  
\_\_\_\_\_  
инициалы, фамилия

## ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 11

Синтаксический разбор текста

по курсу: ОСНОВЫ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. № 1842

/\_\_\_\_\_  
подпись, дата

А.Д. Аничкин  
\_\_\_\_\_  
инициалы, фамилия

Санкт-Петербург 2019

## 1. Цель работы

Закрепление знаний об основных конструкциях языка Си, закрепление навыков самостоятельной разработки алгоритмов и выбора инструментария для их реализации.

## 2. Задание

Задача 11.2

Вывести на экран содержимое файла с программой на языке C, выделяя в строке все целые константы.

## 3. Формализация задачи

Для решения поставленной задачи мы используем функцию `GetStdHandle()` для получения дескриптора окна консоли, с помощью которого, при помощи функции `SetConsoleTextAttribute()` мы будем менять цвет выходящего потока данных на консоль. Обработка файла будет вестись построчно. Целые константы будут выделены красным цветом на чёрном фоне.

## 4. Тестовые примеры

Исходные данные 1:

```
#pragma warning(disable : 4996) // отключение ошибки c4996
#include <stdio.h>
int main(void)
{
    float s; // сумма ряда
    float a; // очередной элемент
    int i; // номер элемента
    int N; // количество проходов
    int z; // очередной элемент знаменателя
    // ввод параметров ряда
    printf("N ="); // вывод подсказки 'N ='
    scanf("%d", &N); // запись значения с клавиатуры в нужном типе данных,
                    // где &(амперсанда) адрес
    // начальное присвоение переменных
    i = 0;
    z = -2;
    s = 0;
    a = 1;
    // цикл
    while (i != N) // цикл с предусловием
    {
        z = z + 3; // увеличиваем дополнительное значение числителя
        a = a * ((float)(2 * i + 1) / z); // рассчитываем очередной элемент
        s = s + a; // составление суммы ряда
        i = i + 1; // увеличиваем порядковый номер элемента
    }
    // вывод результата
    printf("s = %21.20f\n", s); // %x,y(тип данных)

    return 0;
}
```

## Исходные данные 2:

```
#pragma warning(disable : 4996) // отключение ошибки c4996
#include <stdio.h>
#include <conio.h> // библиотека обслуживающая '_getch()' ???
#define YES 1
#define NO 0
int main(void)
{
    // начальное присваивание
    int c;    // текущий символ из потока
    int counter; // количество 'a'
    int flag;  // флажок, по которому мы
               // определяем наличие буквы а в слове
    // начальные присваивания
    counter = 0;
    flag = NO; // изначально обнуляем флажок
    printf("Please, enter some words:\n");
    // цикл чтения символов из потока, связанного с клавиатурой
    while ( (c = getchar()) != EOF )
    {
        if (c == 'a' || c == 'A') // выявляем наличие буквы а
        {
            flag = YES;
        }
        if (c == ' ' || c == '.' || c == '\n' || c == ',')
        {
            if (flag == YES) // проверка на наличие буквы а в слове
            {
                ++counter; // увеличение счётчика на 1
                flag = NO; // сброс флажка
            }
        }
    }
    printf("Number of words that contains letters 'a' = %d\n", counter);
    printf("\nPress any button to close the window\n");
    _getch(); // ожидание ввода чего либо
    return 0;
}
```

## 6. Листинг программы

```
1  #pragma warning(disable : 4996) // отключение ошибки c4996
2  #include <stdio.h>
3  #include <windows.h>
4  #include <locale.h>
5  #define STR 1024
6
7  void processing(char num[STR / 8]);
8
9  void main(void)
10 {
11     char *locale = setlocale(LC_ALL, "");
12     WORD foregroundColor0; // цвет слов
13     WORD textAttribute;    // атрибут текста - цвет символа и фона
14     HANDLE hStdout = GetStdHandle(STD_OUTPUT_HANDLE);
15     foregroundColor0 = FOREGROUND_INTENSITY | FOREGROUND_BLUE
| FOREGROUND_GREEN;
16     textAttribute = foregroundColor0;
17     SetConsoleTextAttribute(hStdout, textAttribute);
18     int i; // счётчик
19     int j; // счётчик
20     char str[STR]; // массив строки
21     char num[STR / 8]; // массив целочисленной константы
22     char *str_ptr = str;
23     FILE *frd;
24     printf("Please enter address of file: ");
25     gets(str); // получаем путь к файлу
26     if (*str_ptr == "")
27     { // удаление кавычек
28         i = 0;
29         while (*(str_ptr + i) != '\0')
30         {
31             (*(str_ptr + i + 1) == "") ? (*(str_ptr + i) = '\0') : (*(str_ptr + i)
= *(str_ptr + i + 1));
32             ++i;
33         }
34     }
35     printf("Path of file: %s\n", str); // отладка
36     frd = fopen((str), "rt");
37     if (frd == NULL)
38         return; // chk open file
39
40     while (!feof(frd)) // for EOF
41     {
42         fgets(str, STR, frd); // get file string
43         str_ptr = str;
44         i = 0;
45         while (*(str_ptr + i) != '\0')
46         {
47             if (*(str_ptr + i) == '/' && *(str_ptr + i + 1) == '/')

```

```

48             { // защита от //
49                 while (*(str_ptr + i) != '\0')
50                 {
51                     printf("%c", *(str_ptr + i));
52                     ++i;
53                 }
54                 break;
55             }
56             if (*(str_ptr + i) == '-' || (*(str_ptr + i) >= '0' && *(str_ptr + i)
<= '9'))
57             {
58                 j = 0;
59                 while (*(str_ptr + i) != '\0' && *(str_ptr + i) != ';' &&
*(str_ptr + i) != ' ' && *(str_ptr + i) != ',' && *(str_ptr + i) != '}' && *(str_ptr + i) !=
']' && *(str_ptr + i) != '\n')
60                 {
61                     num[j] = *(str_ptr + i);
62                     ++i;
63                     ++j;
64                 }
65                 --i;
66                 num[j] = '\0';
67                 processing(num);
68             }
69             else
70                 printf("%c", *(str_ptr + i));
71             ++i;
72         }
73     }
74     fclose(frd);
75     return 0;
76 }
77
78 void processing(char num[STR / 8])
79 {
80     WORD foregroundColor0; // цвет обычный
81     WORD foregroundColor1; // цвет необычный
82     WORD textAttribute; // атрибут текста - цвет символа и фона
83     HANDLE hStdout = GetStdHandle(STD_OUTPUT_HANDLE);
84     foregroundColor0 = FOREGROUND_INTENSITY | FOREGROUND_BLUE
| FOREGROUND_GREEN;
85     foregroundColor1 = FOREGROUND_INTENSITY | FOREGROUND_RED;
86     int j = 0;
87     int value = strlen(num);
88     while (num[j] == '-' || (num[j] >= '0' && num[j] <= '9'))
89         ++j;
90     if (value == j)
91     {
92         textAttribute = foregroundColor1;
93         SetConsoleTextAttribute(hStdout, textAttribute);

```

```
94         printf("%s", num);
95         textAttribute = foregroundColor0;
96         SetConsoleTextAttribute(hStdout, textAttribute);
97     }
98     else
99         printf("%s", num);
100     return;
101 }
```

## 7. Вывод; д-во правильности работы

Программа работает, что подтверждают тестовые примеры и снимки экрана

```
Консоль отладки Microsoft Visual Studio
Please enter address of file: C:\Users\zerko\Desktop\main.c
Path of file: C:\Users\zerko\Desktop\main.c
#pragma warning(disable : 4996) // отключение ошибки c4996
#include <stdio.h>
int main(void)
{
    float s; // сумма ряда
    float a; // очередной элемент
    int i; // номер элемента
    int N; // количество проходов
    int z; // очередной элемент знаменателя
    // ввод параметров ряда
    printf("N = "); // вывод подсказки 'N ='
    scanf("%d", &N); // запись значения с клавиатуры в нужном типе данных,
                    // где &(амперсанда) адрес
    // начальное присвоение переменных
    i = 0;
    z = -2;
    s = 0;
    a = 1;
    // цикл
    while (i != N) // цикл с предусловием
    {
        z = z + 3; // увеличиваем дополнительное значение числителя
        a = a * ((float)(2 * i + 1) / z); // рассчитываем очередной элемент
        s = s + a; // составление суммы ряда
        i = i + 1; // увеличиваем порядковый номер элемента
    }
    // вывод результата
    printf("s = %21.20f\n", s); // %x,y(тип данных)

    return 0;
}
```

```
Консоль отладки Microsoft Visual Studio
Please enter address of file: C:\Users\zerko\Desktop\main1.c
Path of file: C:\Users\zerko\Desktop\main1.c
#pragma warning(disable : 4996) // отключение ошибки c4996
#include <stdio.h>
#include <conio.h> // библиотека обслуживающая '_getch()' ???
#define YES 1
#define NO 0
int main(void)
{
    // начальное присваивание
    int c; // текущий символ из потока
    int counter; // количество 'a'
    int flag; // флажок, по которому мы
              // определяем наличие буквы а в слове
    // начальные присваивания
    counter = 0;
    flag = NO; // изначально обнуляем флажок
    printf("Please, enter some words:\n");
    // цикл чтения символов из потока, связанного с клавиатурой
    while ( ( c = getchar() ) != EOF )
    {
        if ( c == 'a' || c == 'A' ) // выявляем наличие буквы а
        {
            flag = YES;
        }
        if ( c == ' ' || c == '.' || c == '\n' || c == ',' )
        {
            if (flag == YES) // проверка на наличие буквы а в слове
            {
                ++counter; // увеличение счётчика на 1
                flag = NO; // сброс флажка
            }
        }
    }
    printf("Number of words that contains letters 'a' = %d\n", counter);
    printf("\nPress any button to close the window\n");
    _getch(); // ожидание ввода чего либо
    return 0;
}
```