



Отчет по лабораторной работе № 11 по курсу 1

Фундаментальная информатика

Студент группы М8О-112Б-22 Модин-Глазков Б.А., № по списку 20

Контакты www, e-mail: B.glazkov-modin@mail.ru

Работа выполнена: « 1 » ноября 2021 г.

Преподаватель: Никулин Сергей Петрович

Входной контроль знаний с оценкой _____

Отчет сдан « » _____ 2021 г., итоговая оценка _____

Подпись преподавателя _____

1. **Тема:** Обработка последовательности литер входного текстового файла. Простейшие приёмы лексического анализа. Диаграммы состояний и переходов.
2. **Цель работы:** Научиться обрабатывать последовательности литер входного текстового текста.
3. **Задание:** Вариант 13: у всех допустимых 16-ричных чисел перенести ведущие нули в младшие разряды слова фиксированной длины.
4. **Оборудование :**

ЭВМ Intel Pentium G2140, процессор 3.30 GHz, имя узла сети Cameron с ОП 8096 Мб, НМД 7096 Мб.
Терминал Asus адрес dev/pets/3. Принтер
Другие устройства

Оборудование ПЭВМ студента, если использовалось:

Процессор Intel Core i7 2.6 GHz с ОП 16000Мб, НМД 58000 Мб. Монитор Встроенный дисплей Retina(16-дюймовый 3072 x 1920)

Другие устройства

5. **Программное обеспечение:** *Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства Unix, наименование Ubuntu версия 4.15.0
интерпретатор команд bash версия 4.4.20 Система программирования _____ версия _____

Редактор текстов emacs версия 25.2.2

Местонахождение и имена файлов программ и данных: _____

Программное обеспечение ЭВМ студента, если использовалось:

Операционная система семейства Unix, наименование macOS Big Sur версия 11.1

интерпретатор команд bash версия 3.2.57(1)

Система программирования Xcode версия _____

Редактор текстов emacs версия 25.2.2

Местонахождение и имена файлов программ и данных на домашнем компьютере bogdanmodin@mac

6. **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями):

Включаем библиотеки ввода-вывода

Разбираем каждый символ по 3 состоянием(0-где мы считаем "0", 1-докидываем "0" в конец, 2- для всех остальных символов)

При этом в каждом состоянии контролируем

7. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

Код:

```
#include <stdio.h>
int p(char c) {
    return (c == ' ') || (c == '.') || (c == ',') || (c == '\n') || (c == '_');
}
int f1(char c) {
    return (c >= '0' && c <= '9') || (c >= 'a' && c <= 'f');
}
int main () {
    char c;
    int nul = 0, l = 0, i, flag = 0;
    while (flag != 3) {
        c = getchar();
        if (c == EOF) flag = 3;
        else {
            switch(flag) {
                case (0):
                    if (c == EOF) flag = 3;
                    if (p(c)) {
                        putchar(c);
                        break;
                    }
                    if (c == '0') {
                        nul += 1;
                        break;
                    }
                    if (c != '0' && !p(c) && f1(c)) {
                        putchar(c);
                        l += 1;
                        flag = 1;
                        break;
                    }
                    else {
                        putchar(c);
                        flag = 2;
                        break;
                    }
                case (1):
                    if (c == EOF) flag = 3;
                    if (space(c)) {
                        for (i = 0; i < nul; i += 1) {
                            putchar ('0');
                        }
                        nul = 0;
                        l = 0;
                        flag = 0;
                        putchar(c);
                        break;
                    }
                    if (l == 1 && f1(c)) {
                        l += 1;
                        putchar(c);
                        break;
                    }
                    else {
                        putchar(c);
                        l += 1;
                        flag = 2;
                        break;
                    }
                case (2):
                    if (c == EOF) flag = 3;
                    if (p(c)) {
                        nul = 0;
                        l = 0;
                        putchar(c);
                        flag = 0;
                        break;
                    }
                    else {
                        putchar(c);
                        l += 1;
                        break;
                    }
            }
        }
    }
    putchar ('\n');
}
```

Пункты 1-7 отчета составляются строго до начала лабораторной работы.

Допущен к выполнению работы. Подпись преподавателя _____

8. Распечатка протокола (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

```

bogdanmodin@mac Desktop % gcc code.c
bogdanmodin@mac Desktop % ./a.out
43896 ololo feafe 00000000007a fe4210e2 00000010 jojojo yehdhyjiq 0
43896 ololo feafe 7a0000000000 fe4210e2 10000000 jojojo yehdhyjiq 0

```

9. Дневник отладки должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание

10. Замечания автора по существу работы

11. Выводы

В ходе выполнения лабораторной работы я научился работать с типом данных char на языке Си и обрабатывать последовательность символов

Подпись студента _____