**Отчет по лабораторной работе №7**

**Тема**

**Функции ввода-вывода getchar() и putchar()**

*РК6-16Б*

16. Петраков Станислав Альбертович

Преподаватель: Федорук Елена Владимировна

**Цель работы**

Изучение буферизованного ввода-вывода и стандартных функций ввода-вывода getchar() и putchar().

**Ответы на контрольные вопросы**

1. *Что такое буферизованный ввод-вывод?*

Буферизация - способ организации ввода-вывода в программе, позволяющий минимизировать число обращений к устройству. При чтении из дискового файла, блок данных копируется с диска в пользовательский буфер.

1. *Что такое таблица кодирования?*

Это таблица, которой соответствуют символы какому либо номеру. Самые часто используемые ASCII и Unicode.

1. *Для чего и как используется функция getchar()?*

Функция для получения 1 символа из буфера.

1. *Для чего и как используется функция putchar()?*

Для вывода 1 символа в поток.

1. *Что означает EOF?*

EOF (аббревиатура англ. end of file — конец файла) в компьютерной терминологии является индикатором операционной системы, означающим, что данные в источнике закончились. Источниками данных обычно являются файлы и потоки..

1. *Как в языке Си представляется строка?*

Массивом символов.

1. *Каким символом должен оканчиваться массив символов?*

Нулевым байтом \0.

**Программа**

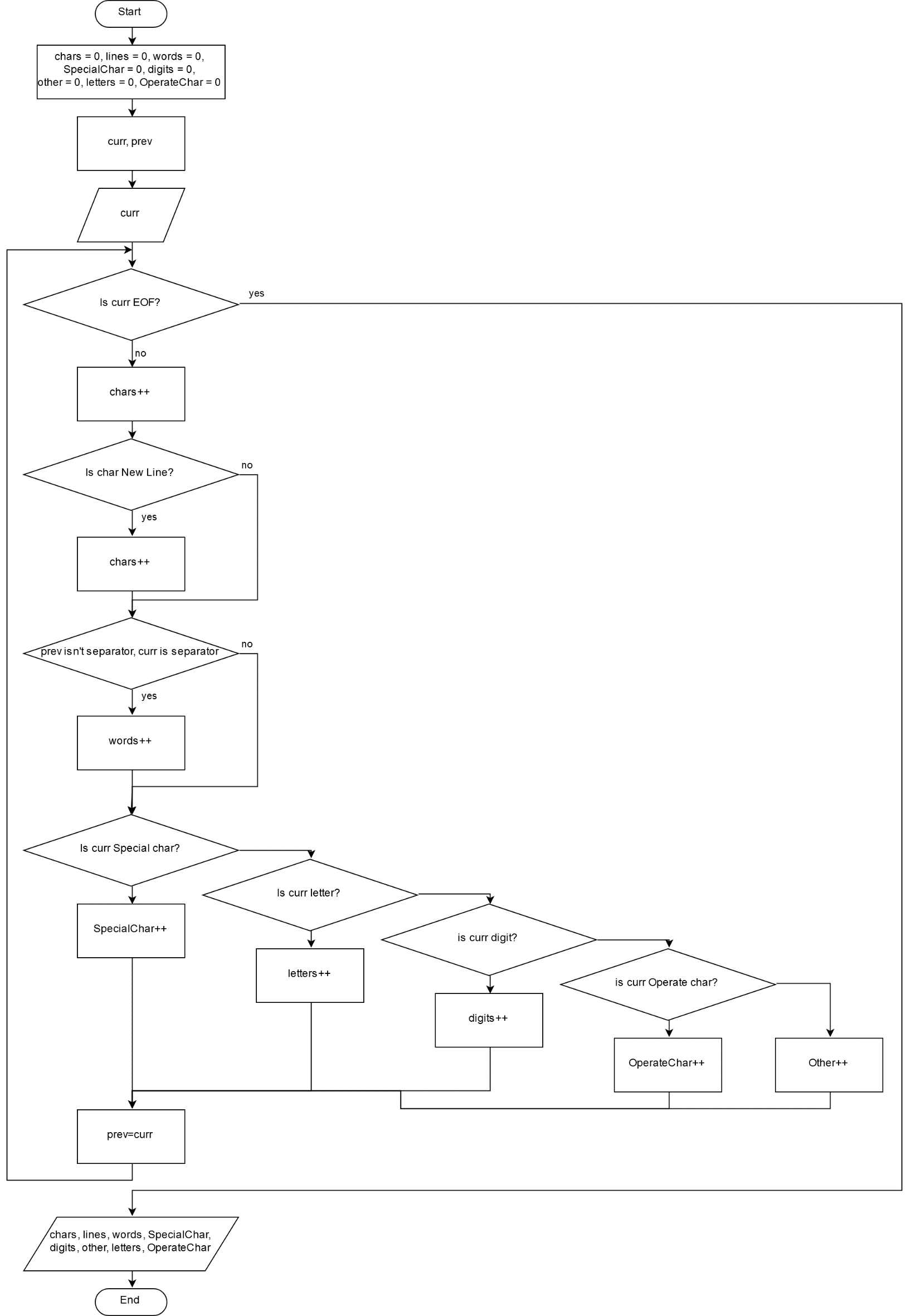
*Задание:*

1. Разработайте программу, которая вводит со стандартного потока ввода множество символов, ограниченное вводом признака конца файла (EOF). Определите и выведите на экран число введенных символов, строк и слов, а также выведите на экран монитора эти символы и строки. Кроме того, подсчитайте количество управляющих символов, букв латинского алфавита, цифр, знаков операций.

2. Выполните эту программу, вводя различные комбинации символов, в том числе символов табуляции, пробела, <Enter>, <CTRL>+<d>. В отчете зафиксируйте результаты и объясните их.

3. Выполните программу, перенаправляя вывод (ввод) в файл (из файла). Для этого используйте команды перенаправления ввода и вывода OC UNIX.

**Блок-схема:**

****

**Сама программа:**

#include <stdio.h>

int main()

{

int chars = 0, lines = 0, words = 0, SpecialChar = 0, digits = 0, other = 0, letters = 0, OperateChar = 0;

int curr = 0, prev = 0;

while ((curr = getchar()) != EOF)

{

chars++;

if (curr == '\n')

lines++;

if ((prev != '\t' && prev != '\n' && prev != ' ') &&

(curr == '\t' || curr == '\n' || curr == ' '))

words++;

if (curr >= 0 && curr <= 31)

SpecialChar++;

else if ((curr >= 'A' && curr <= 'Z') || (curr >= 'a' && curr <= 'z'))

letters++;

else if (curr >= '0' && curr <= '9')

digits++;

else if ((curr >= '!' && curr <= '/') || (curr >= ':' && curr <= '@') ||

(curr >= '[' && curr <= '`') || (curr >= '{' && curr <= '~'))

OperateChar++;

else

other++;

prev = curr;

putchar(curr);

}

printf("\nChars: %d.\n", chars);

printf("Letters: %d.\n", letters);

printf("Lines: %d.\n", lines);

printf("Words: %d.\n", words);

printf("Special Chars: %d.\n", SpecialChar);

printf("Operate Chars: %d.\n", OperateChar);

printf("Digits: %d.\n", digits);

printf("Other: %d.\n", other);

}

**Результаты работы программы**

Программа корректно подсчитывает все нужные символы.

**Вывод**

Мной был изучен буферизированный ввод/вывод и могу использовать функции getchar() и putchar() .