**Отчет по лабораторной работе №7** по курсу Фундаментальная информатика

Студент группы М8О-103Б-23 Зайцев Иван Денисович, № по списку 11

Контакты www, e-mail: mai.zaitsev.ivan@gmail.com

Работа выполнена: «25» ноября 2023г.

Преподаватель: Никулин Сергей Петрович каф. 806

Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Отчет сдан « » \_\_\_\_\_\_\_\_\_201 \_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. **Тема:** Техника работы с множествами в Си. Анализ ввода строк. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Цель работы: Составление программы на языке Си, которая для любых допустимых и корректно записанных строк, поступающих на стандартный ввод программы, выполняет указанные в варианте действия.**
2. **Задание** Написать простую программу на языке си, которая будет проверять, есть ли в тексте слово, включающее в себя только одну гласную возможно несколько раз(19 вариант)
3. **Оборудование *ПЭВМ студента, если использовалось****:* **16 Гб оперативной памяти, экран диагональю 17.3", с разрешением экрана 1920x1080 Пикс**

Процессор  **AMD Ryzen 7 5700U with Radeon Graphics**

1. **\*Программное обеспечение**

Операционная система семейства \_\_\_Unix\_\_\_\_, наименование \_\_Ubuntu\_\_\_\_\_\_\_\_\_\_\_\_ версия \_\_\_22.04\_\_\_

интерпретатор команд \_\_\_\_bash\_\_\_\_\_\_\_\_ версия \_\_5.1.16\_\_\_\_.

Система программирования \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_версия \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Редактор текстов \_\_\_\_\_emacs, gnuplot\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ версия \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Утилиты операционной системы cat; cc; ptype; help; gcc

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Прикладные системы и программы: Microsoft Visual studio code версия 1.83

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Местонахождение и имена файлов программ и данных на домашнем компьютере \_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

1.При помощи while ((c = getchar()) != EOF) будем поочередно считывать литеры, пока не получим EOF

2. Если литера не входит в множество {«,», « », «\t», «\n»}, тогда добавляем ее в множество current word(при помощи объединения), предварительно конвертируя при помощи функции char\_to\_set

3. Если литера входит в множество {«,», « », «\t», «\n»}, значит, что слово у нас уже сформировано и при помощи объединения с множеством vowels в функции crossing (пересечение), мы получаем множество гласных этого слова

4.Дальше выполняя проверку на степень двойки(т.к. множество — битовая маска, в случае если у нас одна гласная, оно будет таковым являться), получаем результат, в зависимости от которого выводим, что слово содержит только один вид гласных и меняем переменную flag, которая будет отвечать за финальный вывод программы, иначе — выводим, что слово не соответствует требуемому условию

Программуможно запустить только с одним тестом, так как отслеживание выполняется по каждому слову и все случаи будут отображены.

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

#include <stdio.h>

#include <limits.h>

#include<ctype.h>

//есть ли слово, содержащее одну гласную\согласную (возможно несколько раз)

unsigned int char\_to\_set(char cc){

cc = tolower(cc);

if (cc < 'a' || cc > 'z'){

return 0;

}

else{

return 1u<<(cc-'a');

}

}

unsigned int adding(unsigned int a, unsigned int b){

return a | b;

}

unsigned int crossing(unsigned int a, unsigned int b){

return a & b;

}

int check(unsigned int word){

while (word % 2 == 0){

word /= 2;

if (word == 0){

return 0;

}

}

if (word == 1){

return 1;

}

else{

return 0;

}

}

int main(){

char c;

unsigned int current\_word = 0;

int flag = 0;

long int co = 1;

unsigned int vowels = char\_to\_set('a') | char\_to\_set('e') | char\_to\_set('u') | char\_to\_set('o') | char\_to\_set('i');

printf("Введите последовательность слов в которых разделитель - пробел, запятая, граница строки, табуляция\n");

while ((c = getchar()) != EOF){

if (c != ' ' & c != ',' & c != '\n' & c != '\t'){

unsigned int current\_letter = char\_to\_set(c);

current\_word = adding(current\_word, current\_letter);

}

else if (current\_word == 0){

continue;

}

else{

if (check(crossing(current\_word, vowels)) == 1 && crossing(current\_word, vowels) != 0){

printf("%lde Слово, написанное вами содержит только одну гласную букву\n", co);

flag = 1;

}

co += 1;

current\_word = 0;

}

}

if (flag == 1){

printf("\nВ тексте есть слова, в которых есть только одна гласная буква\n");

}

else{

printf("\nВ тексте нет слов, в которых присутствует только одна гласная буква\n");

}

return 0;

}

*Пункты 1-7 отчета составляются сторого до начала лабораторной работы.*

*Допущен к выполнению работы.* **Подпись преподавателя\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

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

**ivzaru@ivzaru:~$ cat > lab7.c**

**#include <stdio.h>**

**#include <limits.h>**

**#include<ctype.h>**

**//есть ли слово, содержащее одну гласную\согласную (возможно несколько раз)**

**unsigned int char\_to\_set(char cc){**

**cc = tolower(cc);**

**if (cc < 'a' || cc > 'z'){**

**return 0;**

**}**

**else{**

**return 1u<<(cc-'a');**

**}**

**}**

**unsigned int adding(unsigned int a, unsigned int b){**

**return a | b;**

**}**

**unsigned int crossing(unsigned int a, unsigned int b){**

**return a & b;**

**}**

**int check(unsigned int word){**

**while (word % 2 == 0){**

**word /= 2;**

**if (word == 0){**

**return 0;**

**}**

**}**

**if (word == 1){**

**return 1;**

**}**

**else{**

**return 0;**

**}**

**}**

**int main(){**

**char c;**

**unsigned int current\_word = 0;**

**int flag = 0;**

**long int co = 1;**

**unsigned int vowels = char\_to\_set('a') | char\_to\_set('e') | char\_to\_set('u') | char\_to\_set('o') | char\_to\_set('i');**

**printf("Введите последовательность слов в которых разделитель - пробел, запятая, граница строки, табуляция\n");**

**while ((c = getchar()) != EOF){**

**if (c != ' ' & c != ',' & c != '\n' & c != '\t'){**

**unsigned int current\_letter = char\_to\_set(c);**

**current\_word = adding(current\_word, current\_letter);**

**}**

**else if (current\_word == 0){**

**continue;**

**}**

**else{**

**if (check(crossing(current\_word, vowels)) == 1 && crossing(current\_word, vowels) != 0){**

**printf("%lde Слово, написанное вами содержит только одну гласную букву\n", co);**

**flag = 1;**

**}**

**co += 1;**

**current\_word = 0;**

**}**

**}**

**if (flag == 1){**

**printf("\nВ тексте есть слова, в которых есть только одна гласная буква\n");**

**}**

**else{**

**printf("\nВ тексте нет слов, в которых присутствует только одна гласная буква\n");**

**}**

**return 0;**

**}**

**ivzaru@ivzaru:~$ gcc lab7.c -o coderun**

**ivzaru@ivzaru:~$ ./coderun**

**Введите последовательность слов в которых разделитель - пробел, запятая, граница строки, табуляция**

**tr tr tr tr**

**sdbjajo**

**odsapjw wnkknd dsjbisi kakm**

**8e Слово, написанное вами содержит только одну гласную букву**

**9e Слово, написанное вами содержит только одну гласную букву**

**tytytytyty fdhiejn ewhi**

**m m m m m m m**

**koppop**

**20e Слово, написанное вами содержит только одну гласную букву**

**camapary jbdioja kokjnjk**

**21e Слово, написанное вами содержит только одну гласную букву**

**23e Слово, написанное вами содержит только одну гласную букву**

**В тексте есть слова, в которых есть только одна гласная буква**

**ivzaru@ivzaru:~$**

1. **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. | Дата | Время | Событие | Действие по исправлению | Примечание |
| --- | --- | --- | --- | --- | --- | --- |
|  | или |  |  |  |  |  |
|  | дом. |  |  |  |  |  |
|  |  |  |  |  |  |  |

1. **Замечания автора** по существу работы: Отсутствуют
2. **Вывод по работе:**

В ходе выполнения лабораторной работы я освоил методы работы с множествами, понял, как работает пересечение, объединение, добавление символов во множества

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_