**Отчет по лабораторной работе №13** по курсу фундаментальная информатика

Студентка группы М8О-113Б-22, № по списку 1, Астахова Анастасия Сергеевна

Контакты astakhovaanastasia0201@gmail.com

Работа выполнена: « » \_\_\_\_\_\_\_\_ 2022 г.

Преподаватель: доцент каф. 806 Никулин Сергей Петрович

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

Отчет сдан « » \_\_\_\_\_\_\_\_\_2022 г., итоговая оценка \_\_\_\_\_

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

* 1. **Тема:** Множества \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Цель работы:**  Научиться реализовывать множества в С.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

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

1. **Задание (вариант 2):** Есть ли слово, все согласные которого различны?\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. **Оборудование** (лабораторное):

ЭВМ \_\_\_\_\_\_\_\_\_, процессор \_\_\_\_ \_\_, имя узла сети \_\_\_\_\_\_ с ОП \_\_\_\_\_\_\_\_ Мб, НМД \_\_\_\_\_\_\_ Мб. Терминал \_\_\_\_\_ адрес \_\_\_\_\_\_\_\_\_\_. Принтер \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Другие устройства \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

*Оборудование ПЭВМ студента, если использовалось:*

Ноутбук: Процессор Intel Core i5-10210U с ОП 16 ГБ, SSD 512 ГБ. Другие устройства: мышь Bloody R80

*Компьютер: Процессор Ryzen 5 3600 с ОП 16 ГБ, SSD 512 ГБ. Другие устройства: мышь* Razer BASILISK X HYPERSPEED, [*Logitech G910 (920-008019) Orion Spectrum RGB Mechanical Gaming*](https://www.komus.ru/katalog/tekhnika/kompyuternaya-tekhnika/klaviatury-i-myshi/klaviatury-provodnye/klaviatura-logitech-g910-920-008019-orion-spectrum-rgb-mechanical-gaming/p/659916/?from=n-13776), монитор [*LG UltraGear 27GP750*](https://www.lg.com/ru/monitors/lg-27gp750-b)

1. **Программное обеспечение (лабораторное):**

Операционная система семейства \_\_\_Unix\_\_\_\_, наименование \_\_\_\_\_Ubuntu\_\_\_\_\_\_\_\_\_\_\_\_ версия \_\_4.15.0\_\_\_\_\_\_

интерпретатор команд \_\_\_\_bash\_\_\_\_\_\_\_\_ версия \_\_\_4.4.20\_\_\_\_

Система программирования \_\_С\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ версия \_\_\_\_\_\_\_последняя\_\_\_\_\_\_\_\_\_

Редактор текстов \_\_\_gedit\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ версия \_25.2.2 \_\_\_\_\_\_\_\_\_

Утилиты операционной системы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Прикладные системы и программы \_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Местонахождение и имена файлов программ и данных\_\_stud/208104 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

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

Ноутбук: Операционная система семейства Ubuntu, наименование версия Ubuntu 20.04.3 LTS,

*интерпретатор команд bash версия 5.0.17. Система программирования C. Редактор текстов gedit версия 28.2. Утилиты: стандартные утилиты UNIX. Программное обеспечение: стандартное программное обеспечение системы UNIX.*

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

* Создадим функции, позволяющие работать с новым типом данных set
* В основном теле программы создаем set с согласными
* В цикле while принимаем на вход символы.
* Пока не встретим разделитель, добавляем буквы в сет, если они являются согласными и еще не были добавлены в сет.
* Если мы встретили разделитель, то обнуляем сет.
* Если мы встретили согласную букву, которая уже есть в сете, то меняем флаг на false и в итоге выводим, что повторяющиеся согласные есть.

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

kokodrillo@kokodrillo-VirtualBox:~/work/Лабы\_по\_С$ ./a.out

ne vse soglasnie razlichni

Есть слово, где согласные повторяются

kokodrillo@kokodrillo-VirtualBox:~/work/Лабы\_по\_С$ ./a.out

a vot zdes vse

Во всех словах согласные различны

kokodrillo@kokodrillo-VirtualBox:~/work/Лабы\_по\_С$ ./a.out

abcdefg

Во всех словах согласные различны

kokodrillo@kokodrillo-VirtualBox:~/work/Лабы\_по\_С$ ./a.out

abcbefg abcdefg

Есть слово, где согласные повторяются

*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,подписанныйпреподавателем).

#include <string.h> // Lab 13

#include <stdio.h>

#include <stdbool.h>

#include <ctype.h>

#define n 800000

typedef unsigned set\_data\_elem;

enum {

bits\_per\_char = 8,

bits\_per\_elem = sizeof(set\_data\_elem) \* bits\_per\_char,

datalen = (1 << bits\_per\_char) / bits\_per\_elem

};

typedef struct {

set\_data\_elem data[datalen];

} set;

void set\_clear(set \*s) {

memset(s->data, 0, sizeof(s->data));

}

void set\_insert(set \*s, int c) {

s->data[c / bits\_per\_elem] |= 1u << (c % bits\_per\_elem);

}

void set\_generate(set \*s, bool indicator(int)) {

int i;

set\_clear(s);

for(i = 0; i != 1 << bits\_per\_char; i++) {

if(indicator(i)) {

set\_insert(s, i);

}

}

}

void set\_erase(set \*s, int c) {

s->data[c / bits\_per\_elem] &= ~(1u << c % bits\_per\_elem);

}

bool set\_in(const set \*s, int c) {

return (s->data[c/ bits\_per\_elem] & (1u << c % bits\_per\_elem)) != 0;

}

int set\_size(const set \*s) {

int i, size = 0;

for(i = 0; i != 1 << bits\_per\_char; i++) {

if(set\_in(s, i)) {

size++;

}

}

return size;

}

bool set\_equal(const set \*s1, const set \*s2) {

int i;

for(i = 0; i != datalen; i++) {

if(s1->data[i] != s2->data[i]) {

return false;

}

}

return true;

}

bool set\_includes(const set \*s1, const set \*s2) {

int i;

for(i = 0; i != datalen; i++) {

if((s1->data[i] | s2->data[i]) != s1->data[i]) {

return false;

}

}

return true;

}

set set\_union(const set \*s1, const set \*s2) {

int i;

set result;

for(i = 0; i != datalen; i++) {

result.data[i] = s1->data[i] | s2->data[i];

}

return result;

}

set set\_intersection(const set \*s1, const set \*s2) {

int i;

set result;

for(i = 0; i != datalen; i++) {

result.data[i] = s1->data[i] & s2->data[i];

}

return result;

}

set set\_difference(const set \*s1, const set \*s2) {

int i;

set result;

for(i = 0; i != datalen; i++) {

result.data[i] = s1->data[i] & ~(s2->data[i]);

}

return result;

}

set set\_symmetric\_difference(const set \*s1, const set \*s2) {

int i;

set result;

for(i = 0; i != datalen; i++) {

result.data[i] = s1->data[i] ^ s2->data[i];

}

return result;

}

bool is\_alpha(int c) { return isalpha(c); }

bool is\_digit(int c) { return isdigit(c); }

//--------------------------------------------

bool is\_razd(char a) {

if (a == ' ' || a == '\n' || a == '\t') {

return true;

}

else {

return false;

}

}

int main()

{

set word;

char c;

set consonants;

for (char alpha = 'a'; alpha <= 'z'; alpha++) {

if (alpha != 'a' && alpha != 'e' && alpha != 'i' && alpha != 'o' && alpha != 'u' && alpha != 'y') {

set\_insert(&consonants, alpha);

}

}

bool flag = true;

bool sogl = false;

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

if (is\_razd(c)) {

set\_clear(&word);

}

else if (set\_in(&consonants, c)) {

if (set\_in(&word, c)) {

flag = false;

continue;

}

set\_insert(&word, c);

sogl = true;

}

}

if (flag == false) {

printf("Есть слово, где согласные повторяются\n");

}

else {

printf("Во всех словах согласные различны\n");

}

/\*int i;

set s1,s2;

set\_clear(&s1);

set\_generate(&s1,is\_alpha);

set\_generate(&s2,is\_digit);

set\_insert(&s2,'a');

set\_insert(&s2,'A');

s1=set\_symmetric\_difference(&s1, &s2);

for(i=0;i!=1<<bits\_per\_char;++i)

{

if (set\_in(&s1,i)) printf("%d --- in set (%c)\n",i,i);

// else printf("%d --- not in set\n",i);

}

printf("size set = %d\n",set\_size(&s2));\*/

}

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:

Замечания отсутствуют.

1. **Выводы**

Я научилась пользоваться сетом.

Недочёты при выполнении задания могут быть устранены следующим образом:

Недочеты отсутствуют.

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