

Отчет по лабораторной работе № 13 по курсу Архитектура компьютера и информационных сетей

Студент группы М8О-103Б-22 Клименко Виталий Максимович, № по списку 11

Контакты www, e-mail, icq, skype vitalikklimenko96@gmail.com

Работа выполнена: 26 ноября 2022 г.

Преподаватель: доцент Никулин С.П.

Входной контроль знаний с оценкой _____

Отчет сдан « » _____ 202__ г., итоговая оценка ____

Подпись преподавателя _____

1. **Тема:** Множества _____

2. **Цель работы:** Реализация важнейшего объекта математики - множества _____

3. **Задание (вариант № 13):** Вывести, есть ли в введенном тексте гласная, не входящая ни в одно слово _____

4. **Оборудование (лабораторное):**
ЭВМ _____, процессор _____, имя узла сети _____ с ОП _____ Мб,
НМД _____ Мб. Терминал _____ адрес _____. Принтер _____
Другие устройства _____

Оборудование ПЭВМ студента, если использовалось:

Процессор Intel 4x 3.5GHz _____ с ОП 16 ГБ _____ НМД HDD 200 ГБ _____. Монитор Встроенный 1920x1080
Другие устройства Touchpad Synaptics _____

5. **Программное обеспечение (лабораторное):**
Операционная система семейства _____, наименование _____ версия _____
интерпретатор команд _____ версия _____
Система программирования _____ версия _____
Редактор текстов _____ версия _____
Утилиты операционной системы _____

Прикладные системы и программы _____

Местонахождение и имена файлов программ и данных _____

Программное обеспечение ЭВМ студента, если использовалось:

интерпретатор команд bash _____ версия 5.1.16

Операционная система семейства UNIX _____, наименование Pop!_OS _____ версия 22.04 jammy

Система программирования GNU Compiler Collection (GCC) _____ версия 11.2.0

Редактор текстов Visual Studio Code _____ версия 1.72.2

Утилиты операционной системы _____

Прикладные системы и программы gcc, gdb _____

Местонахождение и имена файлов программ и данных на домашнем компьютере _____

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

Реализовать структуру данных set, несколько базовых функций для нее, выполнить задание

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

1. Реализовать структуру данных set
2. Реализовать несколько базовых функций связанных с множеством
3. При помощи созданных функций выполнить задание

Тесты:

1. predlozhenie iz neskolkikh slov, „njsjnsnsjnsjmm mm m mmmm -> a u
2. auioe ->
3. aoiu -> e
4. hello world! -> a i u
5. i should end this programm ->

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)

```
vitosh@vitosh-pop:~/Studying/mai/labs/l13$ cat makefile
a:
    gcc -Wall -Wextra -o main.out main.c set.c
./main.outvitosh@vitosh-pop:~/Studying/mai/labs/l13$ cat set.c
#include "set.h"

const unsigned long long _vowels = ~((1 << ('a' - 'a')) | (1 << ('e' - 'a')) | (1 << ('i' - 'a')) | (1 << ('o' - 'a')) | (1 << ('u' - 'a')));

void init_set(set* s) {
    s->set = 0;
}

void print_missing_chars(set s) {
    unsigned long long a = s.set;
    int off = 0;

    while (a != 0) {
        if (a % 2 == 0) {
            printf("%c ", off + 'a');
        }
        a /= 2;
        ++off;
    }

    printf("\n");
}

int char_to_set(set* s, char c) {
    unsigned long long pre = s->set;
    if (c < 'a' || c > 'z') {
        return -1;
    }

    s->set = s->set | (1 << (c - 'a'));
    if (pre == s->set) {
        return 0;
    }
    return 1;
}

int read_set(set* s) {
    char c;
    int end = 0;
    while (1) {
        c = getchar();
        if (c == ',' || c == ' ' || c == '\t' || c == '\n') {
            if (c == '\n') return -2;
            break;
        } else if ((int) c == EOF) {
            end = EOF;
            break;
        }
        end += (1 - char_to_set(s, c)) % 2;
    }

    return end;
}

int set_size(set* s) {
    unsigned long long a = s->set;
    int answ = 0;
    while (a != 0) {
        answ += a % 2;
        a /= 2;
    }

    return answ;
}

set set_intersection(set s1, set s2) {
    return set_from(s1.set & s2.set);
}

set set_unification(set s1, set s2) {
    return set_from(s1.set | s2.set);
}

set set_from(unsigned long long a) {
```

```

        set s;
        init_set(&s);
        s.set = a;
        return s;
}
vitos@vitos-pop:~/Studying/mai/labs/l13$ cat set.h
#ifndef SET_H

#define SET_H

#include <stdio.h>
#include <stdlib.h>

typedef struct {
    unsigned long long set;
} set;

void init_set(set* s);

void print_missing_chars(set s);

int char_to_set(set* s, char c);

int read_set(set* s);

int set_size(set* s);

set set_unification(set s1, set s2);

set set_intersection(set s1, set s2);

set set_from(unsigned long long a);

#endif
vitos@vitos-pop:~/Studying/mai/labs/l13$ cat main.c
/* Лабораторная работа №13, вариант 13 */
/* Студент гр. М80-103Б-22 Клименко В. М. */

#include "set.h"

const unsigned long long vowels = ((1 << ('a' - 'a')) | (1 << ('e' - 'a')) | (1 << ('i' - 'a')) | (1 << ('o' -
'a')) | (1 << ('u' - 'a')));

unsigned long long mpow(unsigned long long a, unsigned long long n) {
    unsigned long long exp = 1;
    for (unsigned long long i = 1; i <= n; ++i) {
        exp *= a;
    }
    return exp;
}

void print_bin(unsigned long long a) {
    if (a != 0) {
        print_bin(a / 2);
    }
    printf("%lld", a % 2);
}

int main() {
    set current_set, answer_set, temp;
    init_set(&current_set);
    init_set(&answer_set);
    init_set(&temp);

    printf("Input words:\n");
    int read = 0;

    while (read != EOF) {
        init_set(&current_set);
        read = read_set(&current_set);

        answer_set = set_unification(answer_set, current_set);
        if (read == -2) {
            temp = set_unification(set_intersection(answer_set, set_from(vowels)), set_from(~vowels));
            if (temp.set != -1) {
                printf("Vowels that are not presented in these words: ");
                print_missing_chars(temp);
            } else {
                printf("All the vowels were presented in these words\n");
            }
            init_set(&answer_set);
        }
    }
    printf("End of programm...\n");
}

```

```
        return 0;
    }
vitos@vitos-pop:~/Studying/mai/labs/l13$ make
gcc -Wall -Wextra -o main.out main.c set.c
./main.out
Input words:
predlozhenie iz neskolkikh slov, ,,njsjnsnsjnsjmm mm m mmmm
Vowels that are not presented in these words: a u
auioe
All the vowels were presented in these words
aoiu
Vowels that are not presented in these words: e
hello world!
Vowels that are not presented in these words: a i u
i should end this programm
All the vowels were presented in these words
End of programm...
```

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 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: _____

Недочёты при выполнении задания могут быть устранены следующим образом: _____

Подпись студента _____