

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

Студент группы М8О-103Б-22 Киселев Артём Олегович, № по списку 10

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

Работа выполнена: 2022 г.

Преподаватель: доцент каф. 806 Никулин С.П.

Входной контроль знаний с оценкой _____

Отчет сдан « 9 » декабря 2022 г., итоговая оценка ____

Подпись преподавателя _____

1. Тема: Множества

2. Цель работы: Научиться основам работы со множествами на языке Си

3. Задание (вариант № 12): Есть ли гласная (согласная), входящая в состав всех слов?

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

Оборудование ПЭВМ студента, если использовалось:

Процессор Ryzen 3 3200u 2.6GHz _ с ОП 8 ГБ _ НМД SSD 256 ГБ, HDD 1000 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:

Операционная система семейства UNIX _____, наименование Ubuntu _____ версия 22.04.1

интерпретатор команд bash _____ версия 5.1.16

Система программирования C _____ версия _____

Редактор текстов Visual Studio Code _____ версия 28.2

Утилиты операционной системы _____

Прикладные системы и программы gcc

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)

C - компилируемый статически типизированный язык программирования общего назначения.

gcc - компилятор.

Visual Studio Code - текстовый редактор.

Алгоритм:

1) Начинаем посимвольное считывание введенной строки с помощью цикла:

1.1) Если введен разделительный символ, то делаем объединение множеств(первое изначально состоит только из гласных(согласных), второе множество - временное, оно состоит из прочитанных символов, записываемых в п.1.2)

1.2) Если введена гласная(согласная), то добавляем её во временное множество.

2) Если множество гласных (согласных) после завершения цикла имеет хотябы один символ, то выводим: yes, иначе выводим: no.

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

1) Составить и отладить программу на Си.

2) Сделать протокол.

3) Сделать отчет.

Тесты:

1)Ввод: abc bac

Вывод:

Vowels: yes, 1

Consonants: yes, 2

2) Ввод: abcdefgh abcde abcd ab abbca

Вывод:

Vowels: yes, 1

Consonants: yes, 1

3) Ввод: ab bda jdada ui

Вывод:

Vowels: no

Consonants: no

4)Ввод: abcd abcd acd a

Вывод:

Vowels: yes, 1

Consonants: no

5) Ввод: edfrjshugy aydfsugbh ubhjnoyfads sdfujhil

Вывод:

Vowels: yes, 1

Consonants: yes, 4

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)

```
Labs > lab13 > C 13f.c > is_vowel(int)
1  #include <string.h>
2  #include <stdio.h>
3  #include <stdbool.h>
4  #include <ctype.h>
5
6  typedef unsigned set_data_elem;
7
8  enum {
9      bits_per_char = 8,
10     bits_per_elem = sizeof(set_data_elem) * bits_per_char,
11     datalen = (1 << bits_per_char) / bits_per_elem
12 };
13
14 typedef struct {
15     set_data_elem data[datalen];
16 } set;
17
18 void set_clear(set *s) {
19     memset(s->data, 0, sizeof(s->data));
20 }
21
22 void set_insert(set *s, int c) {
23     s->data[c / bits_per_elem] |= 1u << (c % bits_per_elem);
24 }
25
26 void set_generate(set *s, bool indicator(int)) {
27     int i;
28     set_clear(s);
29     for(i = 0; i != 1 << bits_per_char; i++) {
30         if(indicator(i)) {
31             set_insert(s, i);
32         }
33     }
34 }
35
36 set set_intersection(const set *s1, const set *s2) {
37     int i;
38     set result;
39     for(i = 0; i != datalen; i++) {
40         result.data[i] = s1->data[i] & s2->data[i];
41     }
42     return result;
43 }
44
45 bool is_alpha(int c) { return isalpha(c); }
46 bool is_digit(int c) { return isdigit(c); }
47
```

```

47
48 // Является ли гласным
49 bool is_vowel(int c) {
50     char vowels[] = "AEIOUaeiou";
51     if (strchr(vowels, c) != NULL && c != '\0') {
52         return 1;
53     } else {
54         return 0;
55     }
56 }
57
58 // Является ли согласным
59 bool is_consonant(int c) {
60     char consonants[] = "BCDFGJKLMNPQSTVXZHRWYbcdfgjklnpqstvxzhrwy";
61     if (strchr(consonants, c) != NULL && c != '\0') {
62         return 1;
63     } else {
64         return 0;
65     }
66 }
67
68
69 //Вариант 12
70 // Есть ли гласная (согласная), входящая в состав всех слов?
71 //-----
72 int main()
73 {
74     set temp, vowels, consonants;
75     char symbol;
76     int flag = 1;
77     set_clear(&temp);
78     set_clear(&vowels);
79     set_clear(&consonants);
80     set_generate(&vowels, is_vowel);
81     set_generate(&consonants, is_consonant);
82     while (scanf("%c", &symbol) != EOF) {
83         if ((symbol == ' ' || symbol == ',' || symbol == '\n' || symbol == '\0' || symbol == '\t') && flag == 0) {
84             flag = 1;
85             vowels = set_intersection(&vowels, &temp);
86             consonants = set_intersection(&consonants, &temp);
87             set_clear(&temp);
88         } else if (is_alpha(symbol) != 0) {
89             flag = 0;
90             set_insert(&temp, symbol);
91         }
92     }
93     // Гласные
94     if (set_size(&vowels) > 0) {
95         printf("Vowels: yes, %d\n", set_size(&vowels));
96     } else {
97         printf("Vowels: no\n");
98     }
99     // Согласные
100     if (set_size(&consonants) > 0) {
101         printf("Consonants: yes, %d\n", set_size(&consonants));
102     } else {
103         printf("Consonants: no\n");
104     }
105     return 0;
106 }
107

```

```

• jonaj@JonAJ:~/Desktop/StudyDir/Labs/lab13$ gcc 13f.c -lm -o 13
• jonaj@JonAJ:~/Desktop/StudyDir/Labs/lab13$ ./13
abc bac
Vowels: yes, 1
Consonants: yes, 2
• jonaj@JonAJ:~/Desktop/StudyDir/Labs/lab13$ ./13
abcdefgh abcde abcd ab abbca
Vowels: yes, 1
Consonants: yes, 1
• jonaj@JonAJ:~/Desktop/StudyDir/Labs/lab13$ ./13
ab bda jdada ui
Vowels: no
Consonants: no
• jonaj@JonAJ:~/Desktop/StudyDir/Labs/lab13$ ./13
abcd abcd acd a
Vowels: yes, 1
Consonants: no
• jonaj@JonAJ:~/Desktop/StudyDir/Labs/lab13$ ./13
edfrjshugy aydfsugbh ubhjnoyfads sdfujhil
Vowels: yes, 1
Consonants: yes, 4

```


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: в процессе работы я научился основам работы с множествами, реализованными битовыми операциями, на языке Си, создал программу, которая проверяет есть ли гласная (согласная), входящая в состав всех слов.

Недочёты при выполнении задания могут быть устранены следующим образом:

Подпись студента *АД*