

Отчет по лабораторной работе № 13 по курсу “Фундаментальная информатика”

Студент группы М80-109Б-22 Ефименко Кирилл Игоревич, № 6

Контакты
recrut5678@gmail.com
@vivichv9

Работа выполнена: «20» декабря 2022г.

Преподаватель: каф. 806 Сысоев Максим
Алексеевич

Отчет сдан « _____ » _____ 20__ г., итоговая
оценка _____

Подпись преподавателя

1. Тема: Множества

2. **Цель работы:** Составить программу проверки характеристик введенных последовательностей слов и печати развернутого ответа.
3. **Задание (вариант № номер варианта, если есть):** Вариант 23: Есть ли слово, все согласные которого — звонкие.
4. **Оборудование (студента):**

Процессор AMD ryzen 7 5800U 8x 3.9GH с ОП 16384 Мб, НМД 512Гб. Монитор 1920x1080

5. Программное обеспечение (студента):

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *18.10 cosmic*
интерпретатор команд: *bash* версия *4.4.19*.

Система программирования -- версия --, редактор текстов *emacs* версия *25.2.2*

Утилиты операционной системы --

Прикладные системы и программы --

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)

Создать 2 мн-ва с гласными и согласными(звонкие) буквами. Обработать входной символ, проводя проверку на регистр, буквы, пробел. При нахождении слова, подходящего под условие, программа выводит номер этого слова.

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

Тесты:

```

33 void testIsSpace(){
34     assert(isSpace(c: ' ') == 1);
35     assert(isSpace(c: 'a') == 0);
36 }
37
38 void testToLower(){
39     assert(toLower(c: 'a') == (int) 'a');
40     assert(toLower(c: 'A') == (int) 'a');
41 }
42
43 void testIsLetter(){
44     assert(isLetter(c: 'a') == 1);
45     assert(isLetter(c: '{') == 0);
46 }
47
48 void testAll(){
49     testIsLetter();
50     testIsSpace();
51     testToLower();
52 }
53

```

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

```

1  #include <stdio.h>
2  #include <assert.h>
3
4  int toLower(int c);
5  int isSpace(int c);
6  int isLetter(int c);
7  int charBit(int c);
8
9  void addToSet(unsigned int *s, int c) {
10     *s |= charBit(c);
11 }
12
13 int isKeyExists(unsigned int s, int c) {
14     return ((s & charBit(c)) != 0);
15 }
16
17 int toLower(int c) {
18     return (c >= 'A' && c <= 'Z' ? c - 'A' + 'a' : c);
19 }
20
21 int isSpace(int c) {
22     return (c == ' ' || c == ',' || c == '\n' || c == '\t');
23 }
24
25 int isLetter(int c) {
26     return (c >= 'a' && c <= 'z');
27 }
28
29 int charBit(int c) {
30     return (1 << (c - 'a'));
31 }
32
33 void testIsSpace(){
34     assert(isSpace(c: ' ') == 1);
35     assert(isSpace(c: 'a') == 0);

```

```

43 void testIsLetter(){
44     assert(isLetter(c: 'a') == 1);
45     assert(isLetter(c: '{') == 0);
46 }
47
48 void testAll(){
49     testIsLetter();
50     testIsSpace();
51     testToLower();
52 }
53
54 int main() {
55     testAll();
56
57     int ch, letterFound = 0, isWordFound = 0, isSkip = 0, isInput = 0, num = 0, numFirst = 0;
58     unsigned int soglSet = 0, glasSet = 0;
59
60     addToSet(s: &glasSet, c: 'a');
61     addToSet(s: &glasSet, c: 'o');
62     addToSet(s: &glasSet, c: 'i');
63     addToSet(s: &glasSet, c: 'u');
64     addToSet(s: &glasSet, c: 'y');
65     addToSet(s: &glasSet, c: 'e');
66
67     addToSet(s: &soglSet, c: 'v');
68     addToSet(s: &soglSet, c: 'b');
69     addToSet(s: &soglSet, c: 'g');
70     addToSet(s: &soglSet, c: 'd');
71     addToSet(s: &soglSet, c: 'z');
72     addToSet(s: &soglSet, c: 'l');
73     addToSet(s: &soglSet, c: 'm');
74     addToSet(s: &soglSet, c: 'n');
75     addToSet(s: &soglSet, c: 'r');

```

```

77 while ((ch = getchar()) != '\n') {
78     ch = toLower(c: ch);
79
80     if (!isSpace(c: ch)) {
81         if (isSkip) continue;
82
83         if (!isInput) {
84             num++;
85             isInput = 1;
86         }
87
88         if (isLetter(c: ch)) {
89             if (!letterFound) {
90                 if (!isKeyExists(s: glasSet, c: ch)) {
91                     if (!isKeyExists(s: soglSet, c: ch)) isSkip = 1;
92                     else {
93                         letterFound = 1;
94                         isWordFound = 1;
95                     }
96                 }
97             }
98             else if (!isKeyExists(s: glasSet, c: ch) && !isKeyExists(s: soglSet, c: ch)) isWordFound = 0;
99         }
100     }
101     else {
102         if (isWordFound && !numFirst) numFirst = num;
103
104         isInput = 0;
105         letterFound = 0;
106         isSkip = 0;
107     }
108 }

```

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 других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

| № | Лаб. или дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
|---|---------------------|------------|-------|----------------|-------------------------|-------------|
| 1 | лаб | 12.09.2022 | 11:20 | Не работал EOF | Заменял EOF на „1“ | Мне грустно |

10. Замечания автора по существу работы
Замечаний нет

11. Выводы

Было не интересно писать программу. Работать с множества при помощи битовых операций было очень больно и неприятно.

Думаю, это не очень полезно, так как во всех современных языках множества нормально реализованы.

Недочёты при выполнении задания могут быть устранены следующим образом: --

Подпись студента
