

Отчет по лабораторной работе №13 по курсу фундаментальная информатика

Студент группы М8О-101Б-20 Ядров Артем Леонидович, № по списку 28

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

Работа выполнена: « » _____ 202__ г.

Преподаватель: доцент каф. 806 Никулин Сергей Петрович

Входной контроль знаний с оценкой _____

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

Подпись преподавателя _____

1. Тема: Множества

2. Цель работы: Составить программу проверки характеристик введенных последовательностей слов и печати развернутого ответа в соответствии с вариантом задания.

3. Задание (вариант № 28): Есть ли слово, состоящее только из гласных?

4. Оборудование (лабораторное):

ЭВМ Intel Pentium G2140, процессор 3.30 GHz, имя узла сети Cameron с ОП 8096 Мб, НМД 7906 Мб. Терминал ASUS адрес dev/pets/3 Принтер HP Laserjet 6P
Другие устройства _____

Оборудование ПЭВМ студента, если использовалось:

Процессор Intel core i5 2.60 GHz с ОП 8096 Мб, НМД 131072 Мб. Монитор dell
Другие устройства _____

5. Программное обеспечение (лабораторное):

Операционная система семейства Unix, наименование Ubuntu версия 18.15.0
интерпретатор команд bash версия 4.4.20
Система программирования GNU версия 5.8.13
Редактор текстов emacs версия 25.2.2
Утилиты операционной системы gcc, cat

Прикладные системы и программы _____

Местонахождение и имена файлов программ и данных stud/208104

Программное обеспечение ЭВМ студента, если использовалось:

Операционная система семейства Unix, наименование Fedora версия 32
интерпретатор команд bash версия 5.0.17
Система программирования Clion версия 2020.2.1
Редактор текстов emacs версия 25.2.2
Утилиты операционной системы gcc, cat

Прикладные системы и программы _____

Местонахождение и имена файлов программ и данных на домашнем компьютере home/Temich

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

1. Составим множество гласных латинских букв и назовем его «VOWELS». Также введем 2 состояния: «OUT» (находимся вне слова) и «IN_WORD» (находимся в слове), множество «set», куда мы будем добавлять все считываемые символы. Также обозначим за $c1 = \text{true}$ и $c2 = \text{false}$ логические переменные, первая из которых означает, является ли текущее слово последовательностью букв английского алфавита, а вторая — имеется ли хоть одно слово, состоящее из гласных букв английского алфавита.
2. Считываем очередной символ. Если он не равен концу файла, то возможны 2 варианта. Если текущее состояние «state» равно «OUT», то переходим к пункту 3. В противном случае — к пункту 4.
3. Если текущий символ не является разделителем, то переходим к в состояние «IN_WORD» и выполняем следующий пункт. Если же слово является разделителем, то возвращаемся к пункту 2.
4. Если символ является разделителем, то проверяем, состоит ли слово только из букв (за это отвечает переменная $c1$) и является ли множество «set» подмножеством множества «VOWELS» (разность первого и второго множества равна пустому множеству). Если это верно, то переменной $c2$ присваиваем значение true. Далее обновляем переменные set и $c1$, т. к. слово закончилось. Переходим к пункту 2. Если $c1 = \text{false}$, то смысла проверять это слово нет, т. к. оно уже не состоит только из гласных букв, поэтому перейдем к пункту 2. Если $c1 = \text{true}$, тогда проверяем, является ли символ буквой английского алфавита. Если нет — присваиваем переменной $c1$ значение false. Иначе добавляем в наше множество считанный символ и переходим к пункту 2.
5. После окончания ввода в зависимости от значения переменной $c2$ выведем ответ.

Заметим, что в английском языке буква „у“ может означать гласный и согласный звук. Обработать этот случай можно с помощью добавления переменной, равной предыдущему символу и добавлением еще одной буквы „у“, отвечающую за согласный звук. Если предыдущий символ являлся гласной буквой или разделителем, то добавляем в множество считанных согласную „у“, иначе добавляем «обычную» „у“, которая теперь отвечает лишь за гласный звук.

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

Тесты:

1. Input: It would be correct to call the present era the era of science. Science is everything that surrounds us. Experimental science began hundreds of years ago, when men first started using the energy of fire and water. Being a rational creature, man has explored other mysteries of nature, which has gradually led to many new discoveries.
Output: 39 слово состоит только из гласных букв.
2. Input: CLion is one of the most popular IDE for C/C++ programming language.
Output: В тексте нет слов, состоящих только из гласных букв.

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).

```
[Temich@localhost laabs]$ cat lab13.c
```

```
/* Лабораторная работа №13. Вариант 28.
```

```
 * Проверить, есть ли слово, состоящее только из гласных.
```

```
 * Множества представлены в виде битовой маски.
```

```
 * Студент группы 08-101 А.Ядров*/
```

```
#include <stdbool.h>
```

```
#include <ctype.h>
```

```
#include <stdio.h>
```

```
#define OUT 0
```

```
#define IN_WORD 1
```

```
#define uint unsigned int
```

```
uint char_to_set(char c) {
```

```
    c = tolower(c);
```

```
    if (c < 'a' || c > 'z') {
```

```
        return 0;
```

```
    }
```

```
    return 1u << (c - 'a');
```

```
}
```

```
uint set_union(uint set1, uint set2) {
```

```
    return set1 | set2;
```

```
}
```

```
uint set_intersection(uint set1, uint set2) {
```

```
    return set1 & set2;
```

```
}
```

```
uint negation(uint set) {
```

```
    return ~set;
```

```
}
```

```
uint difference(uint set1, uint set2) {
```

```
    return set_intersection(set1, negation(set2));
```

```
}
```

```
bool subset(uint set1, uint set2) {
```

```
    if (difference(set1, set2) == 0) {
```

```
        return true;
```

```
    }
```

```
    return false;
```

```
}
```

```
bool equal(uint set1, uint set2) {
```

```
    return set1 == set2;
```

```
}
```

```
bool empty(uint set1){
```

```
    return set1 == 0;
```

```
}
```

```
bool set_in(uint set1, int a) {
```

```
    if (set_intersection(set1, 1u << (a - 'a')) != 0) {
```

```
        return true;
```

```
    }
```

```
    return false;
```

```
}
```

```
int main() {
```

```
    bool c1 = true, c2 = false;
```

```
    int c, state = OUT, n = 0, ans, last = ' '; //last - предыдущий символ, ans - номер первого слова, состоящего  
только из гласных
```

```
    uint set = 0;
```

```
    uint VOWELS = (1u << ('a' - 'a') | 1u << ('e' - 'a') | 1u << ('i' - 'a') | 1u << ('o' - 'a') |
```

```

        1u << ('u' - 'a') | 1u << ('y' - 'a'));
while ((c = getchar()) != EOF) {
    switch (state) {
        case OUT: {
            if (c == '\n' || c == '\t' || c == ' ') { //если мы были на разделителе и считали разделитель, то просто
                продолжаем
                last = c;
                continue;
            }
            state = IN_WORD; //переход в другое состояние, если не считали разделитель
            n++; //увеличение номера текущего слова
        }
        case IN_WORD: {
            if (c == '\n' || c == '\t' || c == ' ') {
                if (!c2 && c1 && subset(set, VOWELS)) { //проверка на то, является ли считанное слово
                    подмножеством множества гласных
                    c2 = true;
                    ans = n;
                }
                c1 = true; //обновление переменных для нового слова
                set = 0;
                last = 0;
                state = OUT;
                continue;
            }
            if (c1) {
                if (char_to_set(c) == 0) {
                    c1 = false;
                } else {
                    if (c == 'y' && (set_in(VOWELS, last) || last == ' ' || last == '\n' || last == '\t')) { // обработка 'y'
                        set = set_union(set, 1u << ('z' - 'a' + 1));
                    } else {
                        set = set | char_to_set(c); //добавление символа в множество
                    }
                    last = c;
                }
            }
        }
    }
}
}
}
if (c2) {
    printf("%d слово состоит только из гласных латинских букв\n", ans);
} else {
    printf("В тексте нет слов, состоящих из гласных латинских букв\n");
}
}

```

[Temich@localhost laabs]\$ gcc lab13.c

[Temich@localhost laabs]\$./a.out

It would be correct to call the present era the era of science. Science is everything that surrounds us. Experimental science began hundreds of years ago, when men first started using the energy of fire and water. Being a rational creature, man has explored other mysteries of nature, which has gradually led to many new discoveries.

39 слово состоит только из гласных букв.

[Temich@localhost laabs]\$./a.out

CLion is one of the

most popular

IDE for C/C++

programming language

В тексте нет слов, состоящих из гласных латинских букв

[Temich@localhost laabs]\$

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

10. Замечания автора по существу работы _____

11. Выводы

Я научился составлять программы, написанные на Си, для работы с множествами.

Недочёты при выполнении задания могут быть устранены следующим образом:

Подпись студента _____