



Отчет по лабораторной работе №13 по курсу _____

Студент группы: **M8O-101Б-22, Парфенов Михаил Максимович**, № по списку: **18**, Контакты

mishaslsk@gmail.com Работа выполнена: « _____ » _____ 202__ г.

Преподаватель: **каф. 806 Крылов Сергей Сергеевич**, 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 № 18):
4. **Оборудование**
Оборудование ПЭВМ студента, если использовалось:
Процессор **AMD Ryzen 5 5500U** с ОП **8192 Мб**, ТТН **256 GB**. Мониторы **Dell Monitor, 1920 x 1080, 14 дюймов.**
5. **Программное обеспечение (лабораторное):**
Программное обеспечение ЭВМ студента, если использовалось:
Операционная система семейства **GNU/Linux**, наименование **Arch Linux** версия **5.19.7** интерпретатор команд **zsh** версия 5.8.
Система программирования _версия _
Редактор текстов _ версия _
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)
7. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

Тесты (в скобках пишу ожидаемый вывод):

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).

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

```
int ToLower(int c) {  
    return (c >= 'A' && c <= 'Z') ? c - 'A' + 'a' : c;  
}
```

```
int getint(){  
    int ch = 0;  
    while(1){  
        int c = getchar();  
        c = ToLower(c);  
        if (c == ' ' || c == '\n' || c == EOF){  
            break;  
        }else{  
            ch = ch*10 + (c-'0');  
        }  
    }  
    return ch;  
}
```

```
unsigned int char_to_set(){  
    int c;  
    unsigned int set = 0b0;
```

```

while (1){
c = getchar();
c = ToLower(c);
if(c == ' '){
    continue;
}
else if (c == EOF || c == '\n') {
    break;;
}
else{
    set = set | (1 << (c - 'a'));
}
}
return set;
}

int main() {
int n;
int count_glasnie = 0;
int count_soglasnie = 0;
printf("Enter words ammount: ");
n = getint();
//zyxwvutsrqponmlkjihgfedcba
unsigned int glasnie = 0b01000100000100000100010001;
//zyxwvutsrqponmlkjihgfedcba
unsigned int soglasnie = 0b10111011111011111011101110;
unsigned int set_old = char_to_set();
for (int i = 0; i < n-1; ++i) {
    unsigned int set_new = char_to_set();
    unsigned int res1 = set_old & glasnie;
    unsigned int res2 = set_new & glasnie;
    if ((res1 & res2) != 0) {
        count_glasnie +=1;
    }

    res1 = set_old & soglasnie;
    res2 = set_new & soglasnie;
    if ((res1 & res2) != 0) {
        count_soglasnie +=1;
    }
    set_old = set_new;
}
if (count_glasnie != 0) {
    printf("GLASNIE - YES\n");
} else {
    printf("GLASNIE - N0\n");
}
if (count_soglasnie != 0) {
    printf("SOGLASNIE - YES\n");
} else {
    printf("SOGLASNIE - N0\n");
}
}
}

```

8. **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.		Время	Событие	Действие по исправлению	Примечание

9. Замечания автора по существу работы ____

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

Недочёты при выполнении задания могут быть устранены следующим образом: _

Подпись студента _____