|  |  |
| --- | --- |
|  | **Отчёт по лабораторной работе** № 13  по курсу: 1 фундаментальная информатика  студент группы : M8O-105Б-21 Титеев Рамиль Маратович , № по списку: 23  Адреса www, e-mail, jabber, skype derol.gym@gmail.com  Работа выполнена: “28 ноября 2021г”  Преподаватель: каф. 806 В. К. Титов  Входной контроль знаний с оценкой  Отчёт сдан “ “ 20 г., итоговая оценка  Подпись преподавателя |

1. **Тема**: Множества
2. **Цель работы**: Составить программу проверки характеристик введенных последовательностей слов в   
   соответствии с вариантом задания и печати развернутого ответа.
3. **Задание**:

Вывести все соседние слова, количество различных букв у которых, отличается ровно на 1

1. **Оборудование** (*лабораторное*):

ЭВМ , процессор , имя узла сети с ОП \_ ГБ

НМД \_\_\_ ГБ. Терминал адрес . Принтер

Другие устройства

*Оборудование ПЭВМ студента, если использовалось:*

Процессор Ryzen4600 @ 6x 3.0GHz, ОП 16384 МБ, НМД ГБ. Монитор: встроенный   
Другие устройства

1. **Программное обеспечение** (*лабораторное*):

Операционная система семейства UNIX, наименование: версия \_\_ \_\_

Интерпретатор команд: версия

Система программирования: версия

Редактор текстов: версия

Утилиты операционной системы:

Прикладные системы и программы:

Местонахождения и имена файлов программ и данных:

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства UNIX, наименование Ubuntu версия 20.04

Интерпретатор команд: bash версия

Система программирования: C версия

Редактор текстов: Emacs версия

Утилиты операционной системы:

Прикладные системы и программы:

Местонахождения и имена файлов программ и данных: /usr/bin , a также /bin

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

Текст считывается посимвольно до конца файла. Считывается одно слово. Если предыдущее слово было пустым, то текущее слово копируется в предыдущее и происходит считывание следующего слово.

После того как нашлось два соседних не пустых слова происходит их сравнение. Вычисляются множества этих слов, а затем сравнивается дина этих множеств. Если разница по модулю равна ровно 1, то выводим эту пару. Далее текущее слово снова копируем в предыдущее. И снова начинаем считывание нового слова.

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

***Set.h:***

// for program lab13.cpp

// file Set.h

// --------------------------------------------------

#define N 26

char A='A';

char Z='Z';

char a='a';

char z='z';

char Alph[2\*N]; // alphabet: small and capital letters

int Letter(char c)

{ return c>=a&&c<=z||c>=A&&c<=Z; }

void CreateSet(char W[], char S[])

{ char c; int i;

for(i=0;i<N;i++)S[i]=S[N+i]=0;

for(i=0;c=W[i];i++)

if(c>=a&&c<=z) S[c-a]=1; else

if(c>=A&&c<=Z) S[N+c-A]=1;

}

void DisplaySet(char S[])

{ int i;

Alph[0]=a; Alph[N]=A; Alph[2\*N]=0; // alphabet fills up

for(i=1;i<N;i++)

{ Alph[i]=Alph[i-1]+1;

Alph[i+N]=Alph[i+N-1]+1; } // --"--

printf("\n%s\n",Alph); // and prints

for(i=0;i<2\*N;i++)

printf("%1d",S[i]); printf("\n"); // set prints

}

int EqualSet(char S1[], char S2[])

{ for(int i=0;i<2\*N;i++)

if(S1[i]!=S2[i]) return 0; return 1;

}

void UnionSet(char S1[], char S2[], char S3[])

{ for(int i=0;i<2\*N;i++) S3[i]=S1[i]||S2[i]; }

void IntersSet(char S1[], char S2[], char S3[])

{ for(int i=0;i<2\*N;i++) S3[i]=S1[i]&&S2[i]; }

int IncludeSet(char S1[], char S2[])

{ for(int i=0;i<2\*N;i++)

if(S1[i]==1&&S2[i]==0) return 0; return 1;

}

int EmptySet(char S[])

{ for(int i=0;i<2\*N;i++)

if(S[i]==1) return 0; return 1;

}

void InSet(char c, char S[]) // add to set

{ if(Letter(c))

if(c<=z) S[c-a]=1; else S[c+N-A]=1;

}

***Программа:***

#include <stdio.h>

#include "Set.h"

#define n 30

// Выводит длину сета

int Len(char S[]){

int l = 0;

for(int i = 0; i<2\*N; i++){

if (S[i] == 1){

l++;

}

}

return l;

}

// Сравнение двух сетов

int checker(char W1[], char W2[]){

char cur\_set[N\*2], last\_set[N\*2];

CreateSet(W1, cur\_set); CreateSet(W2, last\_set);

int l1 = Len(cur\_set); int l2 = Len(last\_set);

if ((l1 - l2 == 1) || (l2 - l1 == 1)){

return 1;

}

return 0;

}

// Копирует слово W1 в W2

void copy(char W1[], char W2[]){

for(int i = 0; i<n; ++i){

W2[i] = W1[i];

}

}

// Выводит слово

void PrintWord(char W[]){

char c;

for (int i=0; c = W[i]; i++){

printf("%c", c);

}

printf(" ");

}

int main(){

char s0 = ' ', s=' ', cur\_word[n], last\_word[n];

int c = 0, count = 0;

last\_word[0]='\0';

while (s != EOF){

s = getchar();

if (!Letter(s0) && Letter(s)){

c = 0;

cur\_word[c] = s;

}

else if (Letter(s0) && Letter(s)){

cur\_word[++c] = s;

}

else if (Letter(s0) && !Letter(s)){

cur\_word[++c] = '\0';

if (last\_word[0] != '\0'){

if(checker(cur\_word, last\_word)){

printf("Встретилась следующая пара слов: ");

PrintWord(last\_word); PrintWord(cur\_word);

printf("\n");

count++;

}

}

copy(cur\_word, last\_word);

}

s0 = s;

}

printf ("Всего встретилось %d пар\n", count);

return 0;

}

Test1:

hello world

I am so glad to see you today

can you call me tomorrow

If you want to find Mary, you should to phone her

Test2:

go with me and you win

keep calm and so you can complite the task

Test3:

Mollis dictum et non mattis amet, ut. Malesuada vulputate mollis orci, luctus amet, justo non plate

Id molestie amet, justo consectetur risus amet, ame

Vitae tempus vitae leo, sed in ornare cursus dui nec ex

Test4:

We are real friends

My mum is slim and rather tall

I have long dark hair and brown eyes

He is tall and handsome

We all help our mother and let her have a rest in the evening

Допущен к выполнению работы. Подпись преподавателя

1. **Распечатка протокола** (подклеить листинг окончательного варианта программы с текстовыми примерами, подписанный преподавателем)

(base) ramil@ramil:~/labs/lab 13$ cat header.txt

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Лабараторная работа №13 \*

\* Множества. \*

\* Выполнил студент гр. М8О-105-Б \*

\* Титеев Рамиль Маратович \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(base) ramil@ramil:~/labs/lab 13$ cat lab13.cpp

#include <stdio.h>

#include "Set.h"

#define n 30

// Выводит длину сета

int Len(char S[]){

int l = 0;

for(int i = 0; i<2\*N; i++){

if (S[i] == 1){

l++;

}

}

return l;

}

// Сравнение двух сетов

int checker(char W1[], char W2[]){

char cur\_set[N\*2], last\_set[N\*2];

CreateSet(W1, cur\_set); CreateSet(W2, last\_set);

int l1 = Len(cur\_set); int l2 = Len(last\_set);

if ((l1 - l2 == 1) || (l2 - l1 == 1)){

return 1;

}

return 0;

}

// Копирует слово W1 в W2

void copy(char W1[], char W2[]){

for(int i = 0; i<n; ++i){

W2[i] = W1[i];

}

}

// Выводит слово

void PrintWord(char W[]){

char c;

for (int i=0; c = W[i]; i++){

printf("%c", c);

}

printf(" ");

}

int main(){

char s0 = ' ', s=' ', cur\_word[n], last\_word[n];

int c = 0, count = 0;

last\_word[0]='\0';

while (s != EOF){

s = getchar();

if (!Letter(s0) && Letter(s)){

c = 0;

cur\_word[c] = s;

}

else if (Letter(s0) && Letter(s)){

cur\_word[++c] = s;

}

else if (Letter(s0) && !Letter(s)){

cur\_word[++c] = '\0';

if (last\_word[0] != '\0'){

if(checker(cur\_word, last\_word)){

printf("Встретилась следующая пара слов: ");

PrintWord(last\_word); PrintWord(cur\_word);

printf("\n");

count++;

}

}

copy(cur\_word, last\_word);

}

s0 = s;

}

printf ("Всего встретилось %d пар\n", count);

return 0;

}

(base) ramil@ramil:~/labs/lab 13$ g++ lab13.cpp

(base) ramil@ramil:~/labs/lab 13$ ./a.out < test1

Встретилась следующая пара слов: hello world

Встретилась следующая пара слов: I am

Встретилась следующая пара слов: see you

Встретилась следующая пара слов: call me

Встретилась следующая пара слов: If you

Встретилась следующая пара слов: you want

Встретилась следующая пара слов: Mary you

Всего встретилось 7 пар

(base) ramil@ramil:~/labs/lab 13$ ./a.out < test2

Встретилась следующая пара слов: me and

Встретилась следующая пара слов: keep calm

Встретилась следующая пара слов: calm and

Встретилась следующая пара слов: and so

Встретилась следующая пара слов: so you

Встретилась следующая пара слов: the task

Всего встретилось 6 пар

(base) ramil@ramil:~/labs/lab 13$ ./a.out < test3

Встретилась следующая пара слов: Mollis dictum

Встретилась следующая пара слов: mattis amet

Встретилась следующая пара слов: mollis orci

Встретилась следующая пара слов: orci luctus

Встретилась следующая пара слов: luctus amet

Встретилась следующая пара слов: amet justo

Встретилась следующая пара слов: amet justo

Встретилась следующая пара слов: amet ame

Встретилась следующая пара слов: Vitae tempus

Встретилась следующая пара слов: tempus vitae

Встретилась следующая пара слов: sed in

Встретилась следующая пара слов: ornare cursus

Встретилась следующая пара слов: cursus dui

Встретилась следующая пара слов: nec ex

Всего встретилось 14 пар

(base) ramil@ramil:~/labs/lab 13$ ./a.out < test4

Встретилась следующая пара слов: We are

Встретилась следующая пара слов: are real

Встретилась следующая пара слов: slim and

Встретилась следующая пара слов: hair and

Встретилась следующая пара слов: eyes He

Встретилась следующая пара слов: is tall

Встретилась следующая пара слов: help our

Встретилась следующая пара слов: her have

Встретилась следующая пара слов: in the

Всего встретилось 9 пар

1. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные ошибки (ошибки в сценарии и программе, не стандартные операции) 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 | Дом | 28.11.21 | 11:28 | Выводились не все пары, так как забыл, что разность должна браться по модулю | Добавил условие, что бы проверялась разность для обоих сторон |  |

1. Замечание автора по существу работы \_\_\_\_\_\_\_\_
2. Выводы Я научился программировать на Си, с использованием массива символов, множеств.

Подпись студента