Федеральное агентство связи  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
 «Сибирский государственный университет  
телекоммуникаций и информатики»

Кафедра вычислительных систем

Курсовая работа

по теме «Hangman. Виселица.»

Выполнил:

ст. гр. ИП-213 Ракшун С.Я.

Проверил:

ст. преподаватель Токмашева Е. И.

# Содержание

[Содержание 2](#_Toc135527447)

[Введение и постановка задачи 3](#_Toc135527448)

[Техническое задание 4](#_Toc135527449)

[Описание выполненного проекта 5](#_Toc135527450)

[Выбор слова 5](#_Toc135527451)

[Ввод 5](#_Toc135527452)

[Личный вклад в проект 6](#_Toc135527454)

[Приложение. Текст программы 7](#_Toc135527455)

# Введение и постановка задачи

Цель работы: создать программу для перевода возможности игры в виселицу с программой.

Задачи:

Узнать правила игры виселица

Выбрать способ ввода данных и вывода данных пользователю

Написать программу, играющую с одним словом

Добавить рандомный выбор из множества слов

# Техническое задание

Функциональность проекта: проект позволяет играть в игру Виселица без использования интернета, программ, или других людей.

Формат входных данных: С клавиатуры.

Интерфейс приложения: Приложение работает, пока пользователь играет в игру. Ввод и вывод данных производятся через консоль.  
Для начала программа выбирает рандомное слово из предложенных. Если пользователь вводит верную букву, эта буква появляется в нужном месте слова вместо пробела. Если не угадывает, то соответствующее сообщение выводится пользователю.  
Если пользователь угадывает всё слово, программа поздравляет с победой и завершается. В противном случае, если пользователь потратил все свои попытки, программа показывает всё слово и завершается.

# Описание выполненного проекта

## Выбор слова

В самом начале программа выбирает из файла words.txt случайное слово, которое будет предложено угадать пользователю. Все неизвестные буквы заменены на “\_”.

## Ввод

В каждый ход пользователь вводит с клавиатуры что-то в консоль. Если пользователь ввел букву, которая существует в слове и не была открыта до этого, то пользователю показывается эта буква. В противном случае, выводится сообщение о некорректности введенных данных или сообщение о том, что буквы нет в слове или была уже угадана, при этом показывая текущее состояние виселицы.

# Личный вклад в проект

В данном проекте я занимался всем.

# Приложение. Текст программы

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

#define MAX\_LENGTH 20

#define MAX\_TRIES 6

void display\_hangman(int tries) {

printf("Рисуем виселицу...\n\n");

if (tries == 0) {

printf(" +---+\n");

printf(" |\n");

printf(" |\n");

printf(" |\n");

printf(" |\n");

printf(" |\n");

} else if (tries == 1) {

printf(" +---+\n");

printf(" | |\n");

printf(" |\n");

printf(" |\n");

printf(" |\n");

printf(" |\n");

} else if (tries == 2) {

printf(" +---+\n");

printf(" | |\n");

printf(" O |\n");

printf(" |\n");

printf(" |\n");

printf(" |\n");

} else if (tries == 3) {

printf(" +---+\n");

printf(" | |\n");

printf(" O |\n");

printf(" | |\n");

printf(" |\n");

printf(" |\n");

} else if (tries == 4) {

printf(" +---+\n");

printf(" | |\n");

printf(" O |\n");

printf(" /| |\n");

printf(" |\n");

printf(" |\n");

} else if (tries == 5) {

printf(" +---+\n");

printf(" | |\n");

printf(" O |\n");

printf(" /|\\ |\n");

printf(" |\n");

printf(" |\n");

} else {

printf(" +---+\n");

printf(" | |\n");

printf(" O |\n");

printf(" /|\\ |\n");

printf(" / \\ |\n");

printf(" |\n");

}

printf("\n");

}

int string\_length(char\* string) {

int length = 0;

while (\*string != '\0') {

length++;

string++;

}

return length;

}

void replace\_chars(char\* string, char find, char replace) {

while (\*string != '\0') {

if (\*string == find) {

\*string = replace;

}

string++;

}

}

void load\_random\_word(char\* word) {

FILE\* file = fopen("words.txt", "r");

if (file == NULL) {

printf("Ошибка загрузки файла со словами.\n");

exit(EXIT\_FAILURE);

}

int num\_words = 0;

while (fgets(word, MAX\_LENGTH, file) != NULL) {

num\_words++;

}

rewind(file);

srand(time(NULL));

int random\_index = rand() % num\_words;

for (int i = 0; i < random\_index; i++) {

fgets(word, MAX\_LENGTH, file);

}

fclose(file);

}

void play\_game() {

char word[MAX\_LENGTH];

load\_random\_word(word);

int word\_length = string\_length(word) - 1;

char guessed[word\_length];

memset(guessed, '-', word\_length);

int num\_tries = 0;

printf("Добро пожаловать в игру в виселицу!\n");

printf("У вас есть %d попыток, чтобы отгадать слово: ", MAX\_TRIES);

printf("%s\n", guessed);

while (num\_tries < MAX\_TRIES) {

char guess;

printf("Введите букву: ");

scanf(" %c", &guess);

int correct\_guess = 0;

for (int i = 0; i < word\_length; i++) {

if (word[i] == guess) {

guessed[i] = guess;

correct\_guess = 1;

}

}

if (!correct\_guess) {

num\_tries++;

display\_hangman(num\_tries);

printf("Неверно! Осталось %d попыток.\n", MAX\_TRIES - num\_tries);

} else if (correct\_guess == 0) {

printf("Недопустимый ввод. Попробуйте еще раз.\n");

} else {

printf("Верно! %s\n", guessed);

}

if (strcmp(word, guessed) == 0) {

printf("Вы победили! Загаданное слово было \"%s\".\n", word);

return;

}

}

printf("Вы проиграли. Загаданное слово было \"%s\".\n", word);

}

int main() {

play\_game();

return 0;

}