Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение  
высшего образования   
«Алтайский государственный технический университет им. И. И. Ползунова»

Факультет информационных технологий

Кафедра прикладной математики

Отчет защищен с оценкой \_\_\_\_\_

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Отчет

По расчётному заданию №1

**«Разработка тестирующей программы»**

по дисциплине «Программирование – 1 семестр»

Студент группы ПИ-54 Дворников М. С.

Преподаватель доцент,к.т.н. Егорова Е.В.

Барнаул 2025

ЗАДАНИЕ. Данную работу можно рассматривать как создание простейшей тестирующей системы (программы).

Программа работает в режиме диалога «Пользователь-Программа». Программа задаёт вопрос и проверяет ответ. Общее количество вопросов в тесте равно 5. В итоге максимальный итоговый результат – это 15 баллов, минимальный – 0 баллов.

Следует реализовать 2 режима работы программы. В режиме автора

(администратора) пользователь должен видеть правильный ответ на вопрос, в режиме пользователя – нет. Выбор режима работы должен происходить на начальном этапе работы программы.

В качестве дополнительных работ реализовано следующее:

1. использование массивов;
2. использование;
3. использование датчика случайных;
4. использование;
5. кодирование информации;
6. использование функций.

Пароль от режима администратора «пароль»

Текст программы:

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h> // Стандартный вход/выход

#include <locale.h> // Вывод на русском языке

#include <conio.h> // Задержка экрана

#include <string.h> // Работа со строками

#include <stdlib.h> // Использование рандома

#include <time.h> // Обновление рандома

#include <windows.h> // Очистка консоли

#define PASSWORD "льмкзш"

#define PASS\_LEN 6

#define ASK 5 // Сколько вопросов нужно задать

#define ALL 8 // Сколько всего вопросов

#define LONG 256 // Насколько длинные строчки могут быть в файле

#define UT setlocale(LC\_ALL, "ru\_RU.UTF-8") // Для вывода текста файла

#define RU setlocale(LC\_ALL, "Russian") // Для вывода русских слов

int start(int is\_admin);

char read\_shufled\_questions(char questions[ALL][3][5][LONG]);

int get\_answer(char right[LONG]);

void incode(char\* word, char\* result);

int eq(char\* one, const char\* two);

int main() {

RU;

SetConsoleCP(1251); // Установка кодировки консоли для ввода

int is\_admin = 0;

printf("Введите режим работы:\n0 <- Пользователь\n1 <- Администратор\n");

is\_admin = \_getch() - 48; // код 48 соответствует цифре 0

if (!is\_admin) printf("Выбран режим работы \"Пользователь\"\n");

else if (is\_admin == 1) {

printf("Выбран режим работы \"Администратор\"\n");

printf("Введите пароль:\n");

char word[PASS\_LEN + 1], coded[PASS\_LEN + 1];

scanf("%s", &word);

getchar();

incode(word, coded);

if (eq(coded, PASSWORD)) printf("Верный пароль\n");

else {

printf("Неверный пароль\n");

printf("Выбран режим работы \"Пользователь\"\n");

is\_admin = 0;

}

}

else {

printf("Некорректный ввод. Выбран режим работы \"Пользователь\"\n");

is\_admin = 0;

}

printf("Нажмите Enter, чтобы продолжить\n");

\_getch();

system("cls"); // Очищаем экран для дальнейшей работы

int total\_score; // Результат пользователя

char symb; // Ввод пользователя

int cont = 1; // Продолжать тестирование

do {

RU;

printf("Тест по теме \"Программирование\"\n\n");

printf("Инструкция по вводу ответов:\n");

printf("Если у вопроса есть варианты ответа, то нужно\n");

printf("вводить ответ полностью, а не букву варианта\n");

printf("Например, если вы хотите выбрать вариант \"а) int\"\n");

printf("то следует ввести \"int\"\n\n\n");

total\_score = start(is\_admin);

printf("Ваш результат: %d\n", total\_score);

printf("Нажмите ESC, чтобы закончить тестирование\n");

printf("Нажмите любую другую кнопку, чтобы продолжить\n");

symb = \_getch();

if (symb == 27) cont = 0; // Нажат ESC

else system("cls"); // Очищаем экран для дальнейшей работы

} while (cont);

return 0;

}

int start(int is\_admin) {

// Создаём массив для вопросов и считываем их с файлов

char questions[ALL][3][5][LONG];

read\_shufled\_questions(questions);

int total\_score = 0;

for (int task = 0; task < ASK; task++) { // Номер вопроса

UT;

printf("%d. ", task + 1);

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

// i = 0 -> строчки вопроса

// i = 1 -> варианты ответа

for (int n = 0; n < 5; n++) {

// Каждая i может содержать 5 строчек

printf("%s", questions[task][i][n]);

}

}

printf("\n");

// Выводим правильный ответ, если ты админ

if (is\_admin) {

setlocale(LC\_ALL, "Russian"); // Вывод русских слов

RU; printf("Правильный ответ: ");

UT; printf("%s", questions[task][2][0]);

printf("\n");

}

total\_score += get\_answer(questions[task][2][0]);

printf("\n\n\n");

}

return total\_score;

}

char read\_shufled\_questions(char questions[ALL][3][5][LONG]) {

// Прочитать файл и записать его в questions

// Создаём список от 0 до ALL

int shufled[ALL], shuf;

for (shuf = 0; shuf < ALL; shuf++) {

shufled[shuf] = shuf;

}

// Теперь перемешаем его

srand(time(NULL)); // Обновляем рандом при помощи времени

int num; // Рандомное число на каждой итерации

int take; // take создаётся, чтобы менять местами числа

for (shuf = 0; shuf < ALL; shuf++) {

num = rand() % ALL;

take = shufled[num];

shufled[num] = shufled[shuf];

shufled[shuf] = take;

}

UT;

FILE\* to\_read = fopen("questions.txt", "r"); // Открываем файл

char line[LONG]; // Создаём буфер для чтения строк

int task = 0; // Номер вопроса

int i = 0; // Номер строки вопроса или варианта ответа

int quest\_end = 0; // Формулировка вопроса закончилась

while ((fgets(line, LONG, to\_read) != NULL) && (task < ALL)) {

if (line[0] == '\n') { // Символ "\n" разделяет два вопроса

task++;

i = 0;

quest\_end = 0;

}

else if (quest\_end && (line[0] != '\t')) { // Это ответ на вопрос

strcpy(questions[shufled[task]][2][0], line);

}

else if (line[0] == '\t') { // Варианты ответа начинаются на "\t"

if (quest\_end == 0) { // Это первый вариант ответа

i = 0;

quest\_end = 1;

}

if (line[1] != '-') {

// Если вариант ответа начинается с "-", то его не печатаем

strcpy(questions[shufled[task]][1][i], line);

i++;

}

}

else { // Иначе - это вопрос

strcpy(questions[shufled[task]][0][i], line);

i++;

}

}

fclose(to\_read);

return NULL;

}

int get\_answer(char right[LONG]) {

// Получить ответ пользователя

int score; // Сколько баллов будет начислено за ответ

char symb; // Последний введённый символ

int end; // Пользователь закончил вводить ответ или

int correct; // Введённый ответ верный

RU;

for (score = 3; score > 0; score--) {

end = 0; correct = 1; // Обнуляем флаги

for (int i = 0; (i < LONG) && (end == 0); i++) {

// Будем проверять ответ посимвольно

symb = getchar();

if (symb == '\n') {

end = 1;

}

if (symb != right[i]) {

correct = 0;

}

}

if (correct) {

printf("Это правильный ответ, +%d\n", score);

return score;

}

else printf("Это неправильный ответ, осталось попыток: %d\n", score - 1);

}

return score;

}

void incode(char\* word, char\* result) {

// Шифровка методом Цезаря, шаг равен номеру последней буквы

int a = 'а';

char new\_word[PASS\_LEN + 1];

char letter;

for (int i = 0; i < PASS\_LEN; i++) {

letter = word[i] - a;

letter += word[PASS\_LEN - 1] - a;

letter %= 32;

new\_word[i] = letter + a;

}

strcpy(result, new\_word);

result[PASS\_LEN] = '\0';

}

int eq(char\* one, const char\* two) {

// Проверка, равны ли две строки

for (int i = 0; i <= PASS\_LEN; i++) if (one[i] != two[i]) return 0;

return 1;

}

Содержимое файла questions.txt:

Какое имя носит функция, с которой

начинается выполнение программы?

-

main

Какой из указанных тип данных используется

для работы с беззнаковыми целыми числами?

а) float

б) unsigned int

в) short

г) double

д) char

unsigned int

Что printf("%.3lf", 12.345678)

выведет в терминал?

-

12.346

Какой знак нужно ввести перед

каждым именем переменной в функции scanf?

а) &

б) \*

в) \

г) |

д) =

&

Что будет результатом выполнения операции

"10 / 4", если оба операнда - целые числа?

-

2

Какой оператор в C используется для

получения остатка от деления?

а) &

б) mod

в) /

г) //

д) %

%

Какой размер типа unsigned char в битах?

-

8

Какой спецификатор формата используется

для вывода целых десятичных чисел?

а) %s

б) %e

в) %x

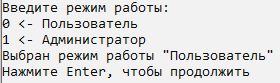
г) /o

д) %d

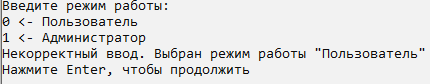
%d

Тесты программы:

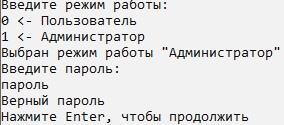
Ввод 0



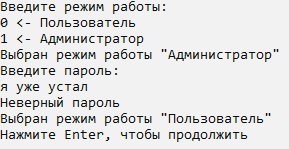
Ввод 2



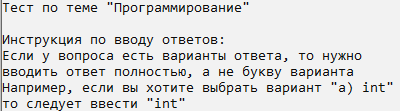
Ввод 1 и верный пароль



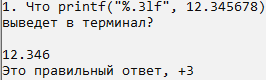
Ввод 1 и неверный пароль

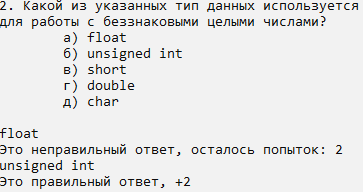


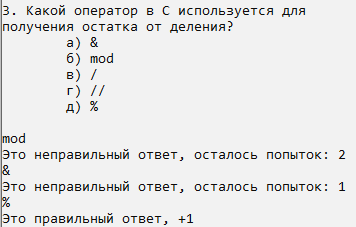
Инструкция после нажатия Enter:

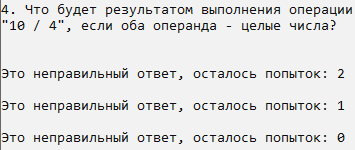


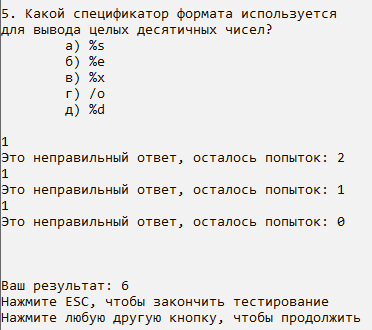
Тесты для пользователя:



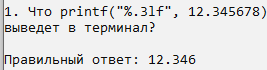




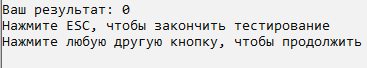




Включённый режим администратора:

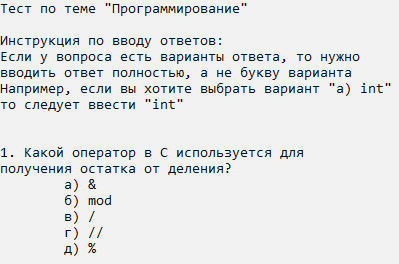


Если нажать ESC



\*Программа завершила работу

Если нажать другую кнопку:



\*Программа начала тест заново