Южно-Уральский государственный колледж, корпус Информационных Технологий и Экономики

**Отчет**

По дисциплине МДК.01.04. Системное программирование

Лабораторная работа №13 «Разработка многофайлового проекта»

Вариант №8

Выполнил студент Яковлев Е.К.

Группа ИСп320Д

2023 г.

**Ход работы:**

1. **Задание:** определить функцию, возвращающую количество минут до окончания суток. Выполнить перегрузку функции для следующих типов параметров:
2. Структура «время» (часы, минуты, секунды).
3. Три целочисленных параметра: часы, минуты, секунды.
4. Два целочисленных параметра: часы, минуты.
   1. Код программы:
5. Файл проекта

int main() {

setlocale(LC\_ALL, "");

struct Time timeFormat3; //Структура "Время" объясленную в файле func.h

while (true)

{

printf("Задайте время в формате 'Часы:Минуты:Cекунды' для структуры: ");

scanf\_s("%d:%d:%d", &timeFormat3.hours, &timeFormat3.minutes, &timeFormat3.seconds); //Задаем значения в структуру

int result1 = MinutesUntilMidnight(timeFormat3); //Функция MinutesUntilMidnight подсчета минут до окончания суток,находится в файле func.cpp и объяслена в файле func.h

printf("Структура время - %d:%d:%d\n", timeFormat3.hours, timeFormat3.minutes, timeFormat3.seconds); //Выводим время из структуры

printf("Минут до окончания дня в формате 'Часы:Минуты:Cекунды': %d\n", result1);

int result2 = MinutesUntilMidnight(timeFormat2(timeFormat3)); //Функция timeFormat2 переводит формат времени из 'чч:мм:сс' в 'чч:мм:' находится в файле func.cpp и объяслена в файле func.h

printf("Минут до окончания дня в формате 'Часы:Минуты': %d\n\n", result2);

}

}

1. Файл func.cpp

#include "func.h"

int MinutesUntilMidnight(struct Time time) {

int minutes = 0;

minutes += (23 - time.hours) \* 60; // Количество минут до полуночи по часам

minutes += (59 - time.minutes); // Количество оставшихся минут

minutes += (60 - time.seconds) / 60; // Корректировка для секунд

return minutes;

}

Time timeFormat2(struct Time timeFormat3) {

struct Time timeFormat2;

timeFormat2.hours = timeFormat3.hours;

timeFormat2.minutes = timeFormat3.minutes;

timeFormat2.seconds = 0;

return timeFormat2;

}

1. Файл func.h

#pragma once

struct Time {

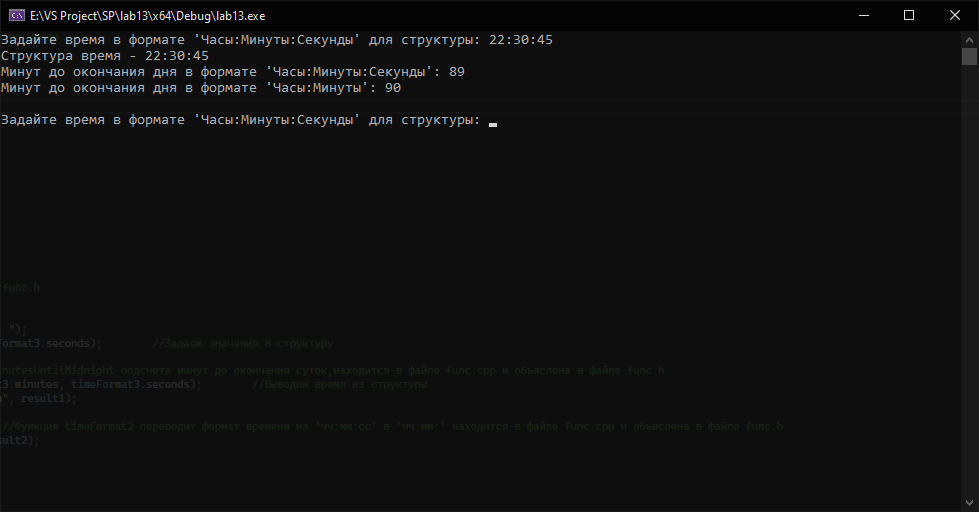
int hours, minutes, seconds;

};

int MinutesUntilMidnight(struct Time time);

Time timeFormat2(struct Time timeFormat3);

* 1. Снимок экрана:



**Контрольные вопросы**

1. **Принципы процедурного программирования.**

Процедурное программирование руководствуется тремя основными принципами: последовательность, выбор и итерация. Принцип последовательности относится к порядку выполнения инструкций сверху вниз в программе. Принцип выбора относится к принятию решений в коде с использованием условных операторов, таких как if , elif и else , или операторов переключения, обеспечивающих альтернативные пути выполнения. Принцип итерации включает конструкции циклов, такие как циклы for и while , которые позволяют выполнять набор инструкций несколько раз с изменениями входных или управляющих параметров.

1. **Компоновка проектов в С/С++.**

Связывает все объектные файлы и статические библиотеки в единый исполняемый файл, который мы и сможем запустить в дальнейшем.

1. **Компиляция проектов, основные этапы.**

В процессе компиляции обычно выделяют следующие подпроцессы (блоки, этапы).

* + - 1. Лексический анализ.
      2. Работа с таблицами.

1. Синтаксический анализ, или разбор.
2. Генерация кода, или трансляция в промежуточный код (например, языка ассемблер).
3. Оптимизация кода.
4. Генерация объектного кода.

В конкретных компиляторах порядок может несколько отличаться, а некоторые блоки могут объединяться в один.

**Вывод:** Таким образом, я научился работать с многофайловыми проектами и написал программу обработки данных в соответствии с заданным вариантом.