Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

**РЕФЕРАТ**

По предмету “Основы алгоритмизации и программирования”

На тему: “Написание таймера на языке программирования С++”

Выполнил:

Студент 1 курса 9 группы

Павлович Ян Андреевич

Преподаватель: Белодед Н.И.

2023, Минск

Содержание

1. Введение
2. Основная часть
   1. Код таймера
   2. Скриншоты работоспособности первой программы
   3. Код таймера как секундомера
   4. Скриншоты работоспособности второй программы
   5. Детальное рассмотрение работы первого таймера
3. Заключение
4. Источники

**Введение**

В данном реферате будет рассмотрена разработка таймера на языке программирования C++. Будет рассмотрена реализация двух вариантов вывода таймера. Оба таймер будут реализованы в консольном окне и будут иметь возможность получать время от пользователя в секундах. После установки времени, первый таймер будет запущен и будет отсчитывать время от заданного значения до нуля. По достижении нулевого значения, на экран вместо таймера будет выведено сообщение "Время вышло!". После установки времени, второй таймер будет запущен и будет отсчитывать время от нуля до заданного времени. По достижении заданного значения, на экран вместо таймера будет выведено сообщение "Достигнуто установленное время!". Код таймера будет представлен с подробными пояснениями, а также будут рассмотрены отдельные части кода, чтобы обеспечить максимальную понятность его работы.

**Код таймера**

#include <iostream>    // Ввод-вывод в консоль

#include <iomanip>     // Для управления форматом вывода

#include <chrono>      // Для работы с временем

#include <thread>      // Для создания задержек

**using** **namespace** std;

// функция для запуска таймера с обратным отсчетом

**void** runTimer(**int** seconds) {

    setlocale(LC\_CTYPE, "Russian"); //добавляем возможность работы с русским языком

**int** totalSeconds = seconds; // Общее количество секунд в таймере

    // Выводим сообщение о запуске таймера

    cout << "Таймер запущен на " << totalSeconds << " секунд. \nОсталось времени: \n";

    // Цикл обратного отсчета

**for** (**int** i = totalSeconds; i > 0; --i) {

**int** hours = i / 3600; // Вычисление часов

**int** minutes = (i % 3600) / 60; // Вычисление минут

**int** secs = i % 60; // Вычисление секунд

        // Используем управление кареткой для перезаписи строки

        cout << setfill('0') << setw(2) << hours << ":"

            << setfill('0') << setw(2) << minutes << ":"

            << setfill('0') << setw(2) << secs << "\r";

        this\_thread::sleep\_for(chrono::seconds(1)); // Задержка на 1 секунду

    }

    cout << "Время вышло!\n"; // Выводим сообщение после завершения таймера

}

**int** main() {

    setlocale(LC\_CTYPE, "Russian"); //добавляем возможность работы с русским языком

**int** userInput;

    // Запрашиваем у пользователя время таймера

    cout << "Введите время таймера в секундах: ";

    cin >> userInput;

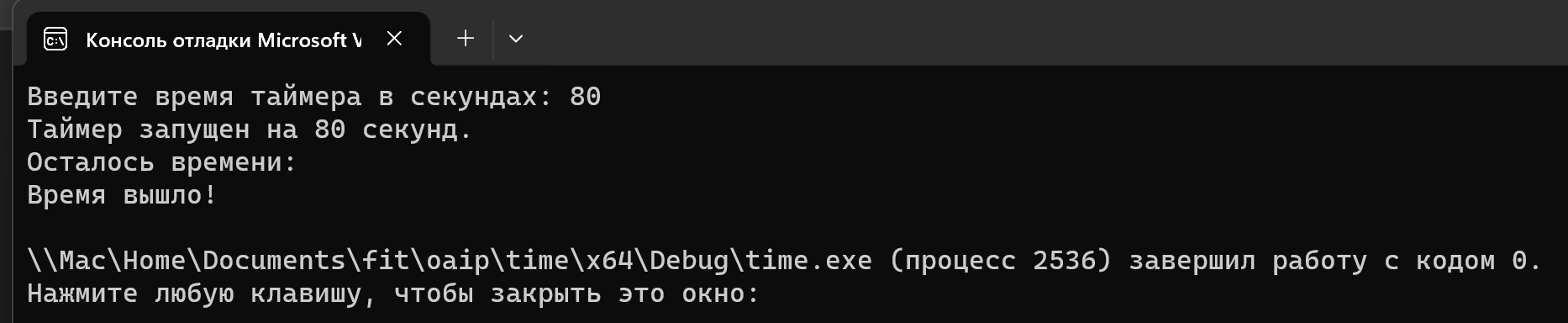
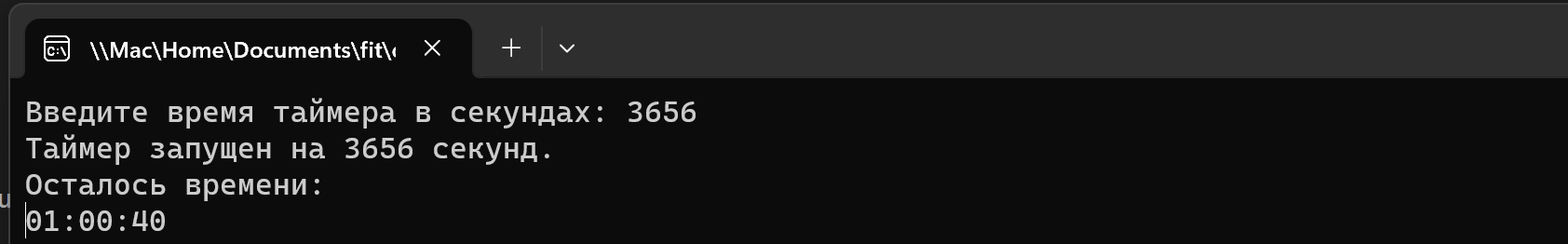
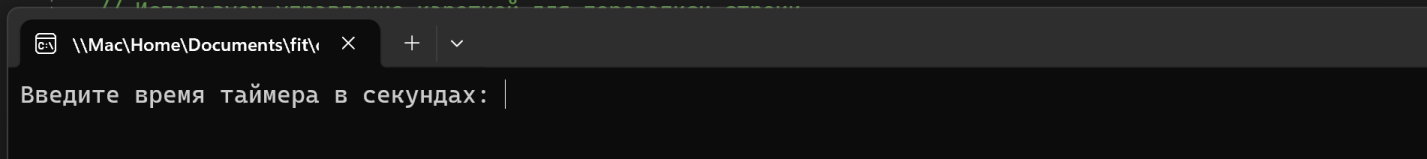
    // Запускаем таймер

    runTimer(userInput);

**return** 0;

}

**Скриншоты работоспособности первой программы**



**Код таймера как секундомера**

#include <iostream>    // Ввод-вывод в консоль

#include <iomanip>     // Для управления форматом вывода

#include <chrono>      // Для работы с временем

#include <thread>      // Для создания задержек

**using** **namespace** std;

// Функция для запуска таймера с обратным отсчетом

**void** runTimer(**int** seconds) {

    setlocale(LC\_CTYPE, "Russian"); //добавляем возможность работы с русским языком

**int** totalSeconds = seconds; // Общее количество секунд в таймере

    // Цикл таймера, в качестве секундомера

**for** (**int** i = 0; i <= totalSeconds; ++i) {

**int** hours = i / 3600; // Вычисление часов

**int** minutes = (i % 3600) / 60; // Вычисление минут

**int** secs = i % 60; // Вычисление секунд

        // Используем управление кареткой для перезаписи строки

        cout << "Прошло времени: " << setfill('0') << setw(2) << hours << ":"

            << setfill('0') << setw(2) << minutes << ":"

            << setfill('0') << setw(2) << secs << "\r";

        this\_thread::sleep\_for(chrono::seconds(1)); // Задержка на 1 секунду

    }

    cout << "Достигнуто установленное время!\n"; // Выводим сообщение после завершения

}

**int** main() {

    setlocale(LC\_CTYPE, "Russian");

**int** userInput;

    // Запрашиваем у пользователя время таймера

    cout << "Введите время таймера в секундах: ";

    cin >> userInput;

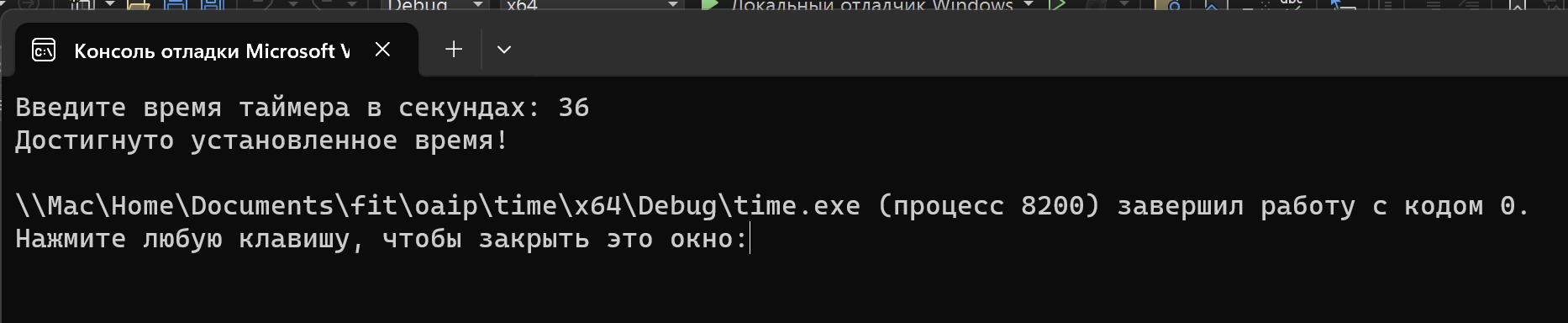
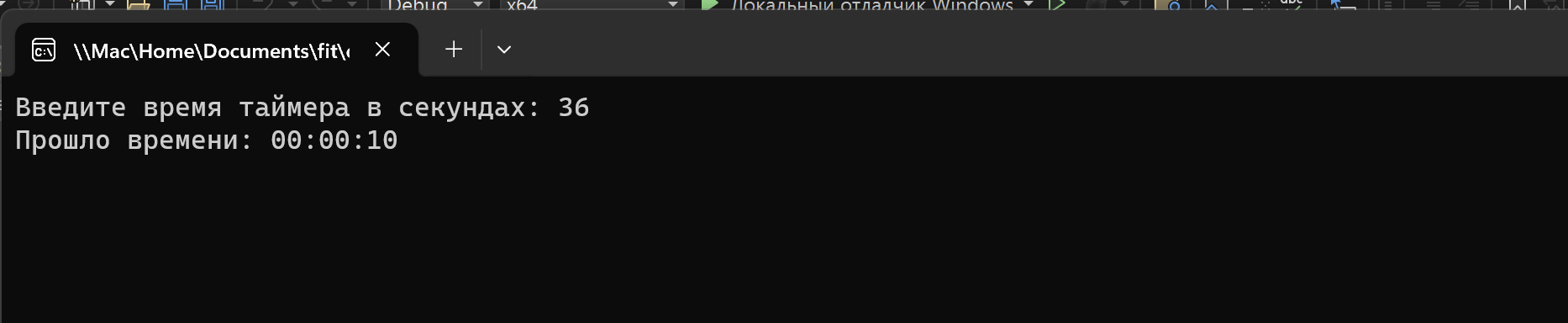
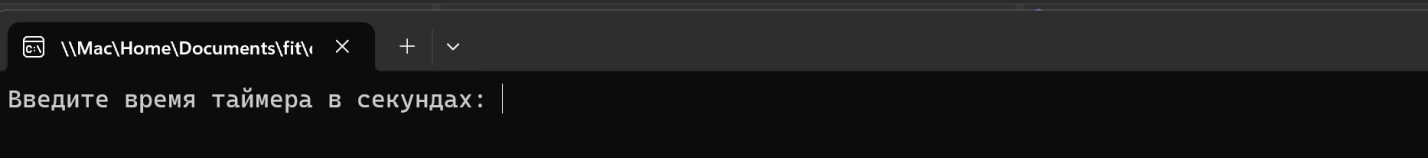
    // Запускаем таймер

    runTimer(userInput);

**return** 0;

}

**Скриншоты работоспособности второй программы**



**Детальное рассмотрение работы первого таймера**

В данном коде представлена функция runTimer, которая запускает таймер с обратным отсчетом. Сначала мы устанавливаем локаль для работы с русским языком, чтобы можно было выводить сообщения на русском языке. Затем мы объявляем переменную totalSeconds, которая содержит общее количество секунд в таймере, и присваиваем ей значение, введенное пользователем.

Далее, мы выводим сообщение о запуске таймера, используя оператор cout, чтобы вывести текст в консоль. В этом сообщении мы указываем общее количество секунд, на которое запущен таймер.

После этого, мы переходим к циклу обратного отсчета. В этом цикле мы вычисляем количество часов, минут и секунд, используя операции деления и остатка от деления. Затем, используя оператор cout и управление кареткой, мы выводим текущее время таймера в формате "часы:минуты:секунды". Для того, чтобы время было выровнено по ширине, мы используем функцию setw из библиотеки iomanip.

В этом коде используется символ возврата каретки (\г), чтобы возвращаться в начало строки и перезаписывать вывод. Таким образом, время остается в той же строке.

После каждой итерации цикла, мы делаем задержку на 1 секунду с помощью функции this\_thread::sleep\_for(chrono::seconds(1)) из библиотеки <thread> в стандартной библиотеке C++. Это позволяет создать паузу между выводом каждого временного значения. Функция std::this\_thread::sleep\_for принимает аргументом объект типа std::chrono::seconds, который представляет собой количество секунд, на которое нужно приостановить выполнение потока. В данном случае, это одна секунда. После выполнения этой строки кода, текущий поток будет приостановлен на одну секунду, что может быть полезно, например, для ожидания завершения какой-то операции или для обеспечения безопасности при работе с многопоточностью.

По завершении цикла, когда таймер достигает значения 0, мы выводим сообщение "Время вышло!".

В функции main мы также устанавливаем локаль для работы с русским языком и запрашиваем у пользователя время таймера в секундах. Затем мы вызываем функцию runTimer, передавая ей значение, введенное пользователем.

Вся программа завершается с возвращаемым значением 0, указывая на успешное выполнение.

**Заключение**

В данном реферате была представлена разработка таймера на языке программирования C++. Были рассмотрены два варианта вывода таймера, оба из которых реализованы в консольном окне и имеют возможность получать время от пользователя в секундах.

Код таймера был представлен с подробными пояснениями, а также были рассмотрены отдельные части кода для обеспечения максимальной понятности его работы.

В результате, данная реализация таймера представляет простой и эффективный способ отслеживания времени в консольном окне. Мы научились разрабатывать таймер с обратным отсчетом на языке программирования C++. Мы использовали оператор cout для вывода сообщений на консоль, функцию setw из библиотеки iomanip для форматированного вывода времени, а также функцию this\_thread::sleep\_for(chrono::seconds(1)) для создания паузы между обновлениями времени. Мы также узнали, как вводить время от пользователя и использовать цикл для обратного отсчета. Весь код был написан на русском языке, и мы использовали локаль для работы с русским языком.

Источники

1. inf-w.ru
2. habr.com
3. studfile.net
4. cyberforum.ru
5. stackoverflow.com
6. ru.wikipedia.org