**Федеральное агентство связи**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования «Московского технического университета связей и информатики»**

Лабораторная работа №1 по дисциплине «Операционные системы»

Выполнил:

Студент группы БВТ1702

Горохов Д.Г.

Москва 2019

**Цель работы**

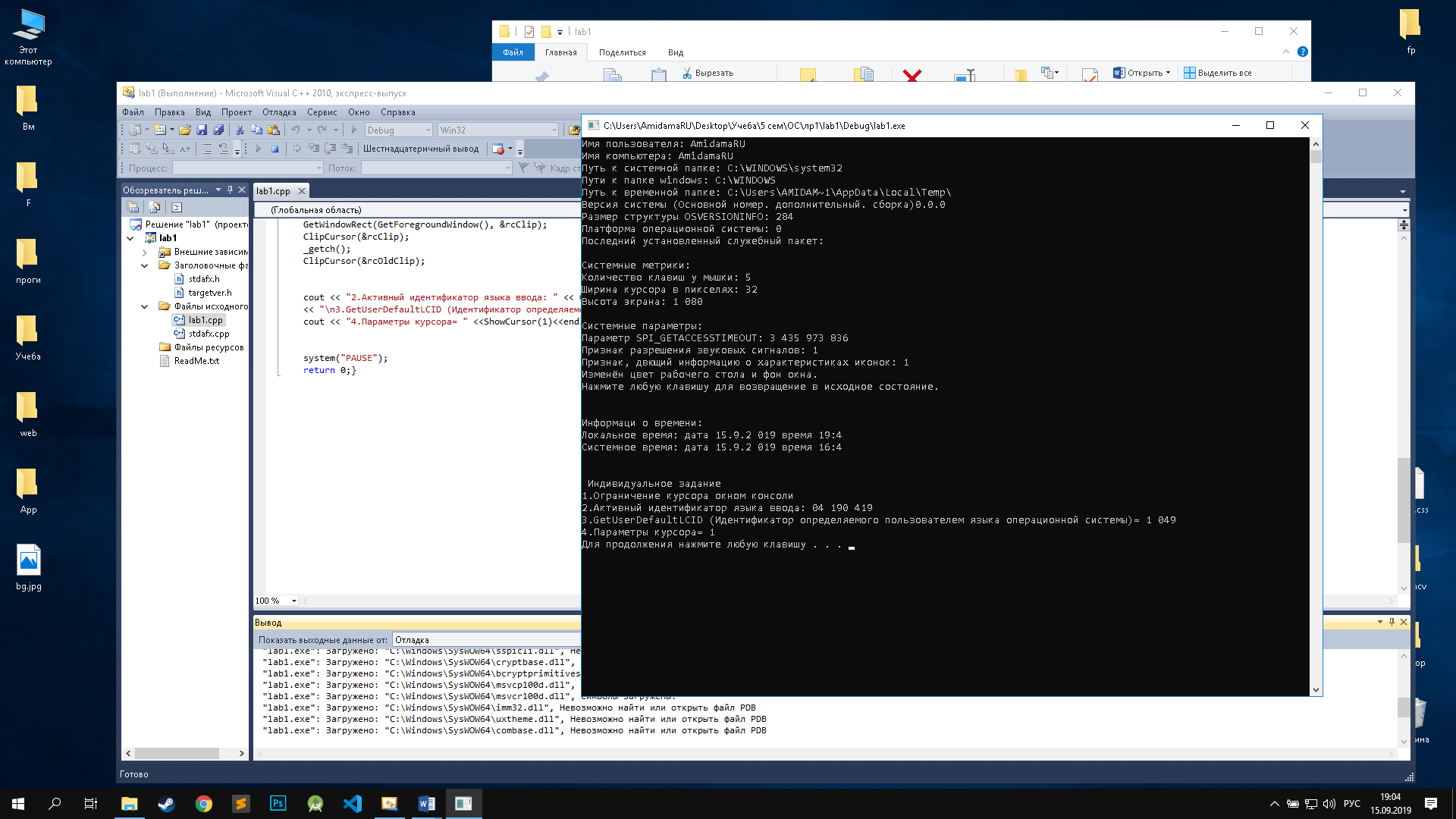
Получение практических навыков по программированию в Win32 API с использованием аппаратных и системных функций.

**Задание**

Разработать программное обеспечение, обеспечивающее получение следующей информации:

* Имя компьютера, имя пользователя;
* Пути к системным каталогам Windows;
* Версия операционной системы;
* Системные метрики (не менее 2 метрик);
* Системные параметры (не менее 2 параметров);
* Системные цвета (определить цвет для следующих символьных констант и изменить его на любой другой) ;
* Функции для работы со временем;
* Дополнительные API-функции.

**Результат работы программы:**



Код программы:

#include "stdafx.h"

#include <iostream>

#include <conio.h>

#include <windows.h>

#include <locale>

#define cout wcout

#define OSVERSIONINFO OSVERSIONINFOEX

using namespace std;

int \_tmain(int argc, \_TCHAR\* argv[])

{

setlocale(LC\_ALL, "rus");

cout.imbue(locale("rus\_rus.1251"));

WCHAR str[4096];

DWORD sz = 4096;

GetUserName(str, &sz);

cout << "Имя пользователя: " << str << endl;

GetComputerName(str,&sz);

cout << "Имя компьютера: " << str << endl;

GetSystemDirectory(str, 4096);

cout << "Путь к системной папке: " << str<< endl;

GetWindowsDirectory(str, 4096);

cout << "Пути к папке windows: " << str<< endl;

GetTempPath(4096, str);

cout << "Путь к временной папке: " << str<< endl;

OSVERSIONINFO osversion;

memset(&osversion, 0, sizeof(OSVERSIONINFO));

osversion.dwOSVersionInfoSize = sizeof(OSVERSIONINFO);

cout << "Версия системы (Основной номер. дополнительный. сборка)" << osversion.dwMajorVersion << "." << osversion.dwMinorVersion << "." << osversion.dwBuildNumber << endl

<< "Размер структуры OSVERSIONINFO: " << osversion.dwOSVersionInfoSize<<endl

<< "Платформа операционной системы: " << osversion.dwPlatformId<<endl

<< "Последний установленный служебный пакет: " << osversion.szCSDVersion<<endl

<< endl << "Системные метрики:" << endl;

cout << "Количество клавиш у мышки: " << GetSystemMetrics(43) << endl;

cout << "Ширина курсора в пикселях: " << GetSystemMetrics(13) << endl;

cout << "Высота экрана: " << GetSystemMetrics(1) << endl;

unsigned int param;

SystemParametersInfo(SPI\_GETACCESSTIMEOUT, 0, &param, 0);

cout << endl<< "Системные параметры:\nПараметр SPI\_GETACCESSTIMEOUT: " << param<< endl;// Временные интервалы

SystemParametersInfo(SPI\_GETBEEP, 0, &param, 0);

cout << "Признак разрешения звуковых сигналов: " << param<< endl;

SystemParametersInfo(SPI\_GETICONMETRICS, 0, &param, 0);

cout << "Признак, двющий информацию о характеристиках иконок: " << param<< endl;

int n[] = {COLOR\_WINDOW, COLOR\_DESKTOP };

COLORREF color = GetSysColor(n[0]);//Dark shadow for three-dimensional display elements.

COLORREF color2 = GetSysColor(n[1]);//Desktop.

unsigned long p[] = { color, color2 };

COLORREF newcolor[] = { RGB(0,60,180), RGB(0,255,0) };//синий //зеленый

SetSysColors(2, n, newcolor);

cout << "Изменён цвет рабочего стола и фон окна.\nНажмите любую клавишу для возвращение в исходное состояние.";

\_getch();

SetSysColors(2, n, p);

cout << endl;

SYSTEMTIME var1, var2;

GetLocalTime(&var1);

GetSystemTime(&var2);

cout << "\n\nИнформаци о времени:\nЛокальное время: дата " << var1.wDay << "." << var1.wMonth << L"." << var1.wYear << " время " << var1.wHour << ":" << var1.wMinute

<< "\nСистемное время: "

<< "дата " << var2.wDay << "." << var2.wMonth << "." << var2.wYear << " время " << var2.wHour << ":" << var2.wMinute<<endl;

cout << "\n\n Индивидуальное задание\n" << "1.Ограничение курсора окном консоли\n";

RECT rcClip;

RECT rcOldClip;

GetClipCursor(&rcOldClip);

GetWindowRect(GetForegroundWindow(), &rcClip);

ClipCursor(&rcClip);

\_getch();

ClipCursor(&rcOldClip);

cout << "2.Активный идентификатор языка ввода: " << GetKeyboardLayout(0)

<< "\n3.GetUserDefaultLCID (Идентификатор определяемого пользователем языка операционной системы)= " << GetUserDefaultLCID()<<endl;

cout << "4.Параметры курсора= " <<ShowCursor(1)<<endl;

system("PAUSE");

return 0;}