

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОРЕНБУРГСКОЙ ОБЛАСТИ  
Государственное автономное профессиональное образовательное  
учреждение  
**«ОРЕНБУРГСКИЙ КОЛЛЕДЖ ЭКОНОМИКИ И ИНФОРМАТИКИ»  
(ГАПОУ ОКЭИ)**

## **Практическая работа**

*ОКЭИ 09.02.07 7026 07 У*

Тема: *«Работа с реестром»*

Выполнил: *Гадияев Ислам Ильгамович*

Оренбург 2026

Содержание

1 Цель работы ..... 3

2 Скриншот структуры готового проекта..... 3

3 Листинг программы ..... 3

4 Результаты работы программы..... 6

					ОКЭИ 09.02.07 7026 07 У									
Изм.	Лист	№ докум.	Подп.	Дата										
Разраб.	Гадилыев И. И.				Практическая работа						Лит.	Лист	Листов	
											У		2	
											Отделение программирования			

## 1 Цель работы

Задание: с помощью winAPI функций для работы с реестром вывести информацию о системе.

## 2 Скриншот структуры готового проекта

На рисунке 1 представлен скриншот структуры готового проекта.

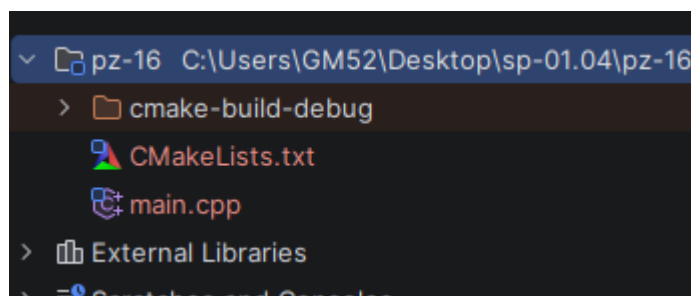


Рисунок 1 – Структура проекта

## 3 Листинг программы

Листинг файла main.cpp представлен ниже.

```
#include <iostream>
#include <windows.h>
#include <string>
#include <vector>

using namespace std;

string ReadRegistryString(HKEY root, const char* path, const char* valueName)
{
    HKEY hKey = nullptr;
    string result = "не удалось прочитать";

    LONG res = RegOpenKeyExA(
        root,
        path,
        0,
        KEY_READ,
        &hKey
    );

    if (res != ERROR_SUCCESS)
    {
        return result + " (ошибка открытия ключа)";
    }

    DWORD dataSize = 0;
    DWORD type = 0;

    RegQueryValueExA(hKey, valueName, NULL, &type, NULL, &dataSize);
```

```

    if (dataSize == 0)
    {
        RegCloseKey(hKey);
        return "значение отсутствует или пустое";
    }

    char* buffer = new char[dataSize];

    res = RegQueryValueExA(
        hKey,
        valueName,
        NULL,
        &type,
        (BYTE*)buffer,
        &dataSize
    );

    if (res == ERROR_SUCCESS && (type == REG_SZ || type == REG_EXPAND_SZ))
    {
        result = buffer;
    }
    else
    {
        result = "ошибка чтения или неверный тип данных";
    }

    delete[] buffer;
    RegCloseKey(hKey);

    return result;
}

void ShowAutoRun(HKEY root, const char* path)
{
    HKEY hKey = nullptr;
    LONG res = RegOpenKeyExA(root, path, 0, KEY_READ, &hKey);

    if (res != ERROR_SUCCESS)
    {
        cout << "Не удалось открыть автозапуск: " << path << endl;
        return;
    }

    cout << "\nПрограммы, запускающиеся автоматически:\n";

    DWORD index = 0;
    char valueName[512];
    DWORD valueNameSize;
    BYTE data[4096];
    DWORD dataSize;
    DWORD type;

    while (true)
    {
        valueNameSize = sizeof(valueName);
        dataSize = sizeof(data);

        res = RegEnumValueA(
            hKey,
            index,
            valueName,
            &valueNameSize,
            NULL,
            &type,
            data,
            &dataSize
        );

        if (res == ERROR_NO_MORE_ITEMS)

```

```

        {
            break;
        }

        if (res == ERROR_SUCCESS && type == REG_SZ)
        {
            cout << valueName << " → " << (char*)data << endl;
        }

        index++;
    }

    RegCloseKey(hKey);

    if (index == 0)
    {
        cout << "(пусто)\n";
    }
}

int main()
{
    SetConsoleOutputCP(CP_UTF8);

    cout << "Операционная система:\n";

    string productName = ReadRegistryString(
        HKEY_LOCAL_MACHINE,
        "SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion",
        "ProductName"
    );
    cout << "Название продукта:      " << productName << endl;

    string releaseId = ReadRegistryString(
        HKEY_LOCAL_MACHINE,
        "SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion",
        "ReleaseId"
    );
    cout << "Выпуск Windows:      " << releaseId << endl;

    string currentBuild = ReadRegistryString(
        HKEY_LOCAL_MACHINE,
        "SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion",
        "CurrentBuild"
    );
    cout << "Номер сборки:      " << currentBuild << endl;

    string systemDir = ReadRegistryString(
        HKEY_LOCAL_MACHINE,
        "SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion",
        "SystemRoot"
    );
    cout << "Системная папка:      " << systemDir << endl;

    cout << "\nBIOS / Производитель:\n";

    string biosVendor = ReadRegistryString(
        HKEY_LOCAL_MACHINE,
        "HARDWARE\\DESCRIPTION\\System\\BIOS",
        "BIOSVendor"
    );
    cout << "Производитель BIOS:      " << biosVendor << endl;

    string biosVersion = ReadRegistryString(
        HKEY_LOCAL_MACHINE,
        "HARDWARE\\DESCRIPTION\\System\\BIOS",
        "BIOSVersion"
    );

```

```

);
cout << "Версия BIOS:          " << biosVersion << endl;

string sysMan = ReadRegistryString(
    HKEY_LOCAL_MACHINE,
    "HARDWARE\\DESCRIPTION\\System\\BIOS",
    "SystemManufacturer"
);
cout << "Производитель системы: " << sysMan << endl;

cout << "\nАвтозапуск (HKEY_LOCAL_MACHINE\\...\\Run): \n";
ShowAutoRun(
    HKEY_LOCAL_MACHINE,
    "SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run"
);

cin.get();

return 0;
}

```

## 4 Результаты работы программы

Результат выполнения программы показан на рисунке 2.

```

C:\Users\GM52\Desktop\sp-01.04\pz-16\cmake-build-debug\pz_16.exe
Операционная система:
Название продукта:   Windows 10 Pro
Выпуск Windows:      2009
Номер сборки:         26100
Системная папка:     C:\WINDOWS

BIOS / Производитель:
Производитель BIOS:   American Megatrends International, LLC.
Версия BIOS:          1.60
Производитель системы: Micro-Star International Co., Ltd.

Автозапуск (HKEY_LOCAL_MACHINE\\...\\Run):

Программы, запускающиеся автоматически:
RtkAudUService -> "C:\WINDOWS\System32\DriverStore\FileRepository\realtekservice.inf_amd64_7e93164ae8ff6bf7\RtkAudUService64.exe" -background

```

Рисунок 2 – Результат работы программы