**Министерство Образования и Науки Кыргызской Республики**

**Кыргызский Государственный Технический Университет им. И.Раззакова**

**ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

Группа: ИВТ-2-13

Предмет: Системное программирование

Вариант: 14

**Лабораторная работа № 2**

**Тема: «Проверка оборудования»**

Выполнил: Жолимбетов С. А

Проверила: Исраилова Н. А.

Бишкек 2016 г.

**Цель работы:** Получение практических навыков в определения конфигурации и основных характеристик компьютера.

**Листинг программы:**

using Microsoft.Win32;

using System;

using System.IO;

using System.Management;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

public static object MessageBox { get; private set; }

static double Round(double num, int e)

{

double r = num \* Math.Pow(10, e);

return (double)Math.Round(r) / Math.Pow(10, e);

}

static string HumanSize(long bytes)

{

double r = Round((double)bytes / (1024 \* 1024 \* 1024), 3);

return r.ToString();

}

static void GetDrive()

{

string volumeinfo;

foreach (DriveInfo drive in DriveInfo.GetDrives())

{

volumeinfo = drive.Name + " " + HumanSize(drive.TotalFreeSpace) + "\t\t" + HumanSize(drive.TotalSize);

Console.WriteLine(volumeinfo);

}

}

static void Main(string[] args)

{

Console.Write("Дата и время: \t\t");

Console.WriteLine(System.DateTime.Now);

Console.Write("Инфо о процессоре: \t");

RegistryKey processorName = Registry.LocalMachine.OpenSubKey(@"Hardware\Description\System\CentralProcessor\0", RegistryKeyPermissionCheck.ReadSubTree);

if (processorName != null && processorName.GetValue("ProcessorNameString") != null)

{

Console.WriteLine(processorName.GetValue("ProcessorNameString"));

}

ManagementObjectSearcher searcher =

new ManagementObjectSearcher("root\\WMI",

"SELECT \* FROM MS\_SystemInformation");

foreach (ManagementObject queryObj in searcher.Get())

{

Console.WriteLine("BIOSVersion: \t\t{0}", queryObj["BIOSVersion"]);

Console.WriteLine("SystemProductName: \t{0}", queryObj["SystemProductName"]);

}

ManagementObjectSearcher battery =

new ManagementObjectSearcher("root\\CIMV2",

"SELECT \* FROM Win32\_Battery");

foreach (ManagementObject queryObj in battery.Get())

{

Console.WriteLine("Caption: \t\t{0}", queryObj["Caption"]);

Console.WriteLine("DeviceID:\t\t{0}", queryObj["DeviceID"]);

Console.WriteLine("EstimatedChargeRemaining:\t{0}", queryObj["EstimatedChargeRemaining"]);

Console.WriteLine("Name: \t\t\t{0}", queryObj["Name"]);

}

ManagementObjectSearcher dm =

new ManagementObjectSearcher("root\\CIMV2",

"SELECT \* FROM Win32\_DesktopMonitor");

foreach (ManagementObject queryObj in dm.Get())

{

Console.WriteLine("MonitorManufacturer: {0}", queryObj["MonitorManufacturer"]);

Console.WriteLine("MonitorType: {0}", queryObj["MonitorType"]);

}

Console.Write("ОС: \t\t\t");

Console.WriteLine(System.Environment.OSVersion);

//Console.Write("Keyboard: \t\t\t");

GetDrive();

Console.ReadKey();

}

}

}

**Результат:**

