МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И

ТЕЛЕКОММУНИКАЦИЙ

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

**Отчёт**

**по лабораторной работе №1**

**по дисциплине**

**«Сетевые технологии»**

Выполнил: студент гр. пм-51 Бакулин А.В. \_\_\_\_\_\_\_\_\_\_\_

Проверил: преподаватель Прозорова Т.Г. \_\_\_\_\_\_\_\_\_\_\_

Киров 2014

**Задание**

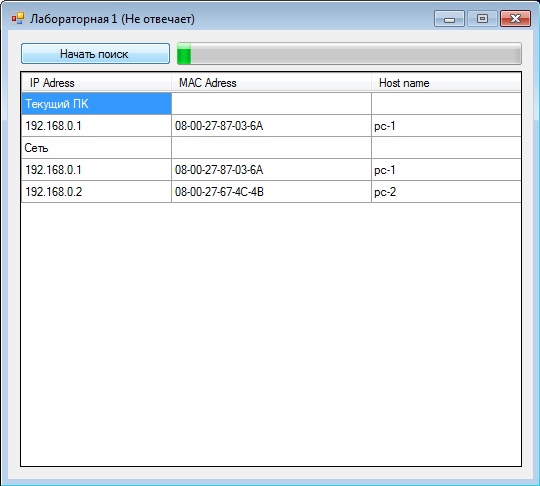
Создать приложение, определяющее все компьютеры (IP-адрес, MAC-адрес и имя), находящиеся в данной сети.

Для нахождения компьютеров в сети можно использовать несколько подходов:

1. компьютеры определяются путѐм посылки сообщения на все возможные IP-адреса;

2. компьютеры находятся с использованием маски.

В первом случае можно получить информацию о компьютерах, не входящих в данную подсеть. Кроме того, такой поиск может занять большой промежуток времени. Во втором случае поиск будет ограничен небольшим диапазоном IP-адресов (для случая лабораторной работы - 254), что позволит избежать выдачи информации по компьютерам, находящимся в другой подсети, а кроме того, позволит получить искомый результат за кратчайший промежуток времени.

****

**Листинг**

namespace task1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

int timeout = 100;

IPAddress myip;

Ping ping = new Ping();

string data = "Hello";

byte[] buffer = Encoding.ASCII.GetBytes(data);

PingReply reply;

MAC mac = new MAC();

string myname = Dns.GetHostName();

IPAddress[] web = Dns.GetHostByName(myname).AddressList;

myip = web[0];

progressBar1.Value = 0;

dataGridView1.Rows.Add();

dataGridView1.Rows[0].Cells[0].Value = "Текущий ПК";

dataGridView1.Rows.Add();

dataGridView1.Rows[1].Cells[0].Value = myip.ToString();

dataGridView1.Rows[1].Cells[2].Value = myname;

dataGridView1.Rows[1].Cells[1].Value = mac.GetMAC(myip);

dataGridView1.Rows.Add();

dataGridView1.Rows[2].Cells[0].Value = "Сеть";

dataGridView1.Refresh();

foreach (IPAddress ip in web)

{

string myipstr = ip.ToString().Remove(ip.ToString().LastIndexOf('.')) + ".";

for (int i = 1; i < 255; i++)

{

progressBar1.Value = (int)(i / 2.54);

reply = ping.Send(myipstr + i, timeout, buffer);

if (reply.Status == IPStatus.Success)

{

dataGridView1.Rows.Add();

myip = reply.Address;

dataGridView1.Rows[dataGridView1.RowCount - 1].Cells[0].Value = myip.ToString();

dataGridView1.Rows[dataGridView1.RowCount - 1].Cells[1].Value = mac.GetMAC(myip);

try

{

IPHostEntry local = Dns.GetHostByAddress(myip);

dataGridView1.Rows[dataGridView1.RowCount - 1].Cells[2].Value = local.HostName;

}

catch (SocketException)

{

dataGridView1.Rows[dataGridView1.RowCount - 1].Cells[2].Value = " - ";

}

dataGridView1.Refresh();

}

}

}

}

}

public class MAC

{

[DllImport("iphlpapi.dll", ExactSpelling = true)]

public static extern int SendARP(int DestIP, int SrcIP, [Out] byte[] pMacAddr, ref int PhyAddrLen);

public string GetMAC(IPAddress ip)

{

byte[] ab = new byte[6];

int len = ab.Length;

int r = SendARP(ip.GetHashCode(), 0, ab, ref len);

return BitConverter.ToString(ab, 0, 6);

}

}

**Вывод**

В результате выполнения лабораторной работы реализован второй подход к нахождению IP и MAC адресов компьютеров в локальной сети.