

## LAB 4

### I. Lý thuyết

- Sử dụng các lớp hỗ trợ lập trình socket của C#
- Gửi và nhận dữ liệu nhị phân

### II. Bài tập:

1) Viết lại các bài tập của Lab 2 và Lab 3 sử dụng các lớp TcpClient, TcpListener và UdpClient.

2) Chương trình gửi và nhận thông tin nhân viên

- Bước 1: Tạo lớp nhân viên

```
class Employee
{
    public int EmployeeID;
    private int LastNameSize;
    public string LastName;
    private int FirstNameSize;
    public string FirstName;
    public int YearsService;
    public double Salary;
    public int size;
}
```

- Bước 2: Bổ sung phương thức chuyển các thành phần của lớp thành mảng bytes

```
public byte[] GetBytes()
{
    byte[] data = new byte[1024];
    int place = 0;
    Buffer.BlockCopy(BitConverter.GetBytes(EmployeeID), 0, data, place, 4);
    place += 4;
    Buffer.BlockCopy(BitConverter.GetBytes(LastName.Length), 0, data, place, 4);
    place += 4;
    Buffer.BlockCopy(Encoding.ASCII.GetBytes(LastName), 0, data, place, LastName.Length);
    place += LastName.Length;
    Buffer.BlockCopy(BitConverter.GetBytes(FirstName.Length), 0, data, place, 4);
    place += 4;
    Buffer.BlockCopy(Encoding.ASCII.GetBytes(FirstName), 0, data, place, FirstName.Length);
    place += FirstName.Length;
    Buffer.BlockCopy(BitConverter.GetBytes(YearsService), 0, data, place, 4);
    place += 4;
    Buffer.BlockCopy(BitConverter.GetBytes(Salary), 0, data, place, 8);
    place += 8;
    size = place;
    return data;
}
```

- Bước 3: Bổ sung các phương thức khởi tạo của lớp (chuyển dữ liệu từ mảng bytes lưu vào các thành phần của lớp

```
public Employee()
{
}
public Employee(byte[] data)
{
    int place = 0;
    EmployeeID = BitConverter.ToInt32(data, place);
    place += 4;
    LastNameSize = BitConverter.ToInt32(data, place);
    place += 4;
    LastName = Encoding.ASCII.GetString(data, place, LastNameSize);
    place = place + LastNameSize;
    FirstNameSize = BitConverter.ToInt32(data, place);
    place += 4;
    FirstName = Encoding.ASCII.GetString(data, place, FirstNameSize);
    place += FirstNameSize;
    YearsService = BitConverter.ToInt32(data, place);
    place += 4;
    Salary = BitConverter.ToDouble(data, place);
}
```

- Bước 4:Chương trình phía Client (gửi dữ liệu)

```
public static void Main()
{
    Employee emp1 = new Employee();

    emp1.EmployeeID = 1;
    emp1.LastName = "Nguyen";
    emp1.FirstName = "Van A";
    emp1.YearsService = 12;
    emp1.Salary = 3500000;

    TcpClient client;
    try
    {
        client = new TcpClient("127.0.0.1", 9050);
    }
    catch (SocketException)
    {
        Console.WriteLine("Khong ket noi duoc voi server");
        return;
    }
}
```

```
        NetworkStream ns = client.GetStream();
        byte[] data = emp1.GetBytes();
        int size = emp1.size;
        byte[] packsize = new byte[2];
        Console.WriteLine("Kich thuoc goi tin = {0}", size);
        packsize = BitConverter.GetBytes(size);
        ns.Write(packsize, 0, 2);
        ns.Write(data, 0, size);
        ns.Flush();

        ns.Close();
        client.Close();
    }
}
```

- Bước 5: Chương trình phía Server (nhận dữ liệu và xuất thông tin)

```
static void Main(string[] args)
{
    byte[] data = new byte[1024];
    TcpListener server = new TcpListener(IPAddress.Any, 9050);
    server.Start();
    TcpClient client = server.AcceptTcpClient();
    NetworkStream ns = client.GetStream();

    byte[] size = new byte[2];
    int recv = ns.Read(size, 0, 2);
    int packsize = BitConverter.ToInt16(size, 0);
    Console.WriteLine("Kich thuoc goi tin = {0}", packsize);
    recv = ns.Read(data, 0, packsize);
    Employee emp1 = new Employee(data);
    Console.WriteLine("emp1.EmployeeID = {0}", emp1.EmployeeID);
    Console.WriteLine("emp1.LastName = {0}", emp1.LastName);
    Console.WriteLine("emp1.FirstName = {0}", emp1.FirstName);
    Console.WriteLine("emp1.YearsService = {0}", emp1.YearsService);
    Console.WriteLine("emp1.Salary = {0}\n", emp1.Salary);

    ns.Close();
    client.Close();
    server.Stop();
}
}
```

***Yêu cầu nâng cao:***

- a) Chỉnh sửa chương trình phía client: cho phép nhập dữ liệu của lớp từ bàn phím, khi nhập xong gửi lên server và hỏi người dùng có tiếp tục không, nếu trả lời “**Khong**” thì thoát chương trình. Server nhận dữ liệu và xuất ra màn hình.
- b) Chỉnh sửa chương trình phía server: ngoài việc xuất dữ liệu ra màn hình còn ghi vào file (.txt) (*Hướng dẫn*: sử dụng lớp StreamWriter).
- 3) Viết lại bài tập 2) bằng cách sử dụng giao thức UDP.