**HỌC PHẦN LẬP TRÌNH ỨNG DỤNG**

**BÀI TẬP THỰC HÀNH SỐ 2**

**I. Thông tin chung:**

* **Mã số bài tập : BT-LTUD-02**
* **Hình thức nộp bài : Nộp qua Moodle môn học**
* **Thời hạn nộp bài : … / … / ……**
* **Nội dung : Chương 2: Nền tảng C# cơ bản**

**Chuẩn đầu ra cần đạt:**

**L.O.3** Sử dụng các Control để thiết kế giao diện của chương trình

**L.O.4** Viết code đúng chuẩn

1. **CẤU TRÚC RẼ NHÁNH**

# Bài 1: Viết chương trình giải phương trình bậc nhất ax + b = 0

**Bài 2: Viết chương trình giải phương trình bậc 2.**

Bài 3: **Xây dựng ứng dụng Tìm số lớn nhất giữa 3 số**

**Bài 4 Viết chương trình chào theo giờ**

Tìm câu chào bằng tiếng Anh tương ứng với giờ được nhập. Biết rằng

•          Giờ từ 0h đến trước 4h: Good night!

•          Giờ từ 4h đến 12h: Good morning!

•          Giờ từ sau 12h đến 17h: Good afternoon!

•          Giờ từ sau 17h đến 20h: Good evening!

•          Sau 20h đến 23h: Good night!

* Xuất câu chào ra màn hình

**Bài 5: Viết chương trình nhập vào năm dương lịch và xuất ra năm âm lịch tương ứng.**

**Biết răng: Năm âm=Can+Chi**

**Can=Năm dương%10;**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| **Canh** | **Tân** | **Nhâm** | **Quý** | **Giáp** | **Ất** | **Bính** | **Đinh** | **Mậu** | **Kỷ** |

**Chi=Năm dương%12**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **Thân** | **Dậu** | **Tuất** | **Hợi** | **Tý** | **Sửu** | **Dần** | **Mẹo** | **Dần** | **Tỵ** | **Ngọ** | **Mùi** |

1. **CẤU TRÚC LẶP**

**Bài 1.** Viết chương trình cho phép nhập vào một số n và in ra màn hình giá trị [-n..n]

**Yêu cầu:** Viết chương trình với mỗi trường hợp sử dụng vòng lặp FOR , WHILE và DO WHILE.

**Bài 2:** Viết chương trình cho phép nhập vào một số n và in ra màn hình số **chẵn**từ [-n..n]

**Yêu cầu:** Viết chương trình với mỗi trường hợp sử dụng vòng lặp FOR , WHILE và DO WHILE.

**Bài 3:** Viết chương trình nhập vào số n và kiểm tra xem n có phải số nguyên tố hay không?

1. **MẢNG MỘT CHIỀU**

Bài 1. Nhập vào dãy số, các số ngăn cách nhau bởi dấy phẩy. In ra:

Dãy số thực: 5, 6.5, 4.2,7,4,8

Làm sao lấy từng số để mình xử lý;

Tách dãy số thành mảng số thực

Trong C# có hỗ trợ hàm tách không?—(Quy luật: chuỗi mình nhập phải ngăn cách bởi ký hiệu: , ;/,..

Split: Dùng để tách chuỗi thành 1 mảng chuỗi theo dấu ngăn cách. Kết quả trả về là 🡪Mảng chuỗi.

* Xử lý số: Lấy từng phần tử trong mảng chuỗi chuyển sang dạng số.

Xây dựng hàm:

Input: chuỗi số nhập vào

Output: mảng số thực

1. Tổng các số thực âm và tổng số thực dương;
2. Tổng số thực lớn hơn X và tổng số thực nhỏ hơn X;
3. Tổng các số nguyên chẵn và các số nguyên lẻ;
4. Tìm số thực âm lớn nhất và số thực dương nhỏ nhất

Hướng dẫn

1. Cách tách số

double[] TachDaySo()

{

try

{

string strDaySo = txtDaySo.Text;

string[] separator = new string[] { "," };

string[] arrDaySo = strDaySo.Split(separator ,

StringSplitOptions.RemoveEmptyEntries);

double[] daySo = new double[arrDaySo.Length];

for (int i = 0; i < arrDaySo.Length; i++)

{

daySo[i] = Convert.ToDouble(arrDaySo[i]);

}

return daySo;

}

catch(Exception ex)

{

MessageBox.Show("Hay kiem tra lai day so ban vua nhap. Error: " + ex.Message);

return null;

}

}

1. Tổng các số thực âm

double[] daySo = TachDaySo();

double tong = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] < 0)

{

tong = tong + daySo[i];

}

1. Tổng các số thực dương

double[] daySo = TachDaySo();

double tong = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] > 0)

{

tong = tong + daySo[i];

}

}

1. Tổng các số thực lớn hơn X

double[] daySo = TachDaySo();

double x = Convert.ToDouble(txtSoX.Text);

double tong = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] > x)

{

tong = tong + daySo[i];

}

}

1. Tổng các số thực nhỏ hơn X

double[] daySo = TachDaySo();

double x = Convert.ToDouble(txtSoX.Text);

double tong = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] < x)

{

tong = tong + daySo[i];

}

}

1. Tổng các số nguyên chẵn

double[] daySo = TachDaySo();

double tong = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] % 2 == 0)

{

tong = tong + daySo[i];

}

}

1. Tổng các số nguyên lẻ

double[] daySo = TachDaySo();

double tong = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] % 2 == 1 || daySo[i] % 2 == -1)

{

tong = tong + daySo[i];

}

}

1. Tìm số thực âm lớn nhất

double[] daySo = TachDaySo();

double max = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] < 0)

{

if (daySo[i] > max || max == 0)

{

max = daySo[i];

}

}

}

1. Tìm số thực dương nhỏ nhất

double[] daySo = TachDaySo();

double min = 0;

for (int i = 0; i < daySo.Length; i++)

{

if (daySo[i] > 0)

{

if (daySo[i] < min || min == 0)

{

min = daySo[i];

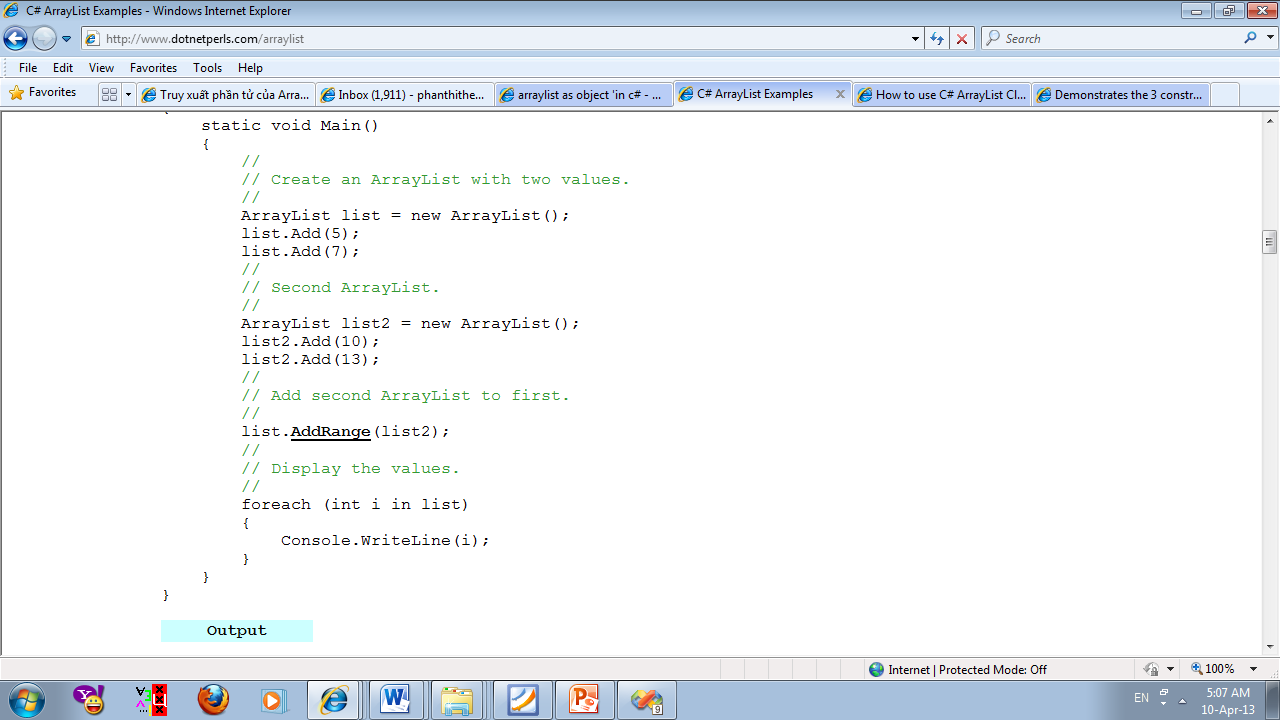
}

}

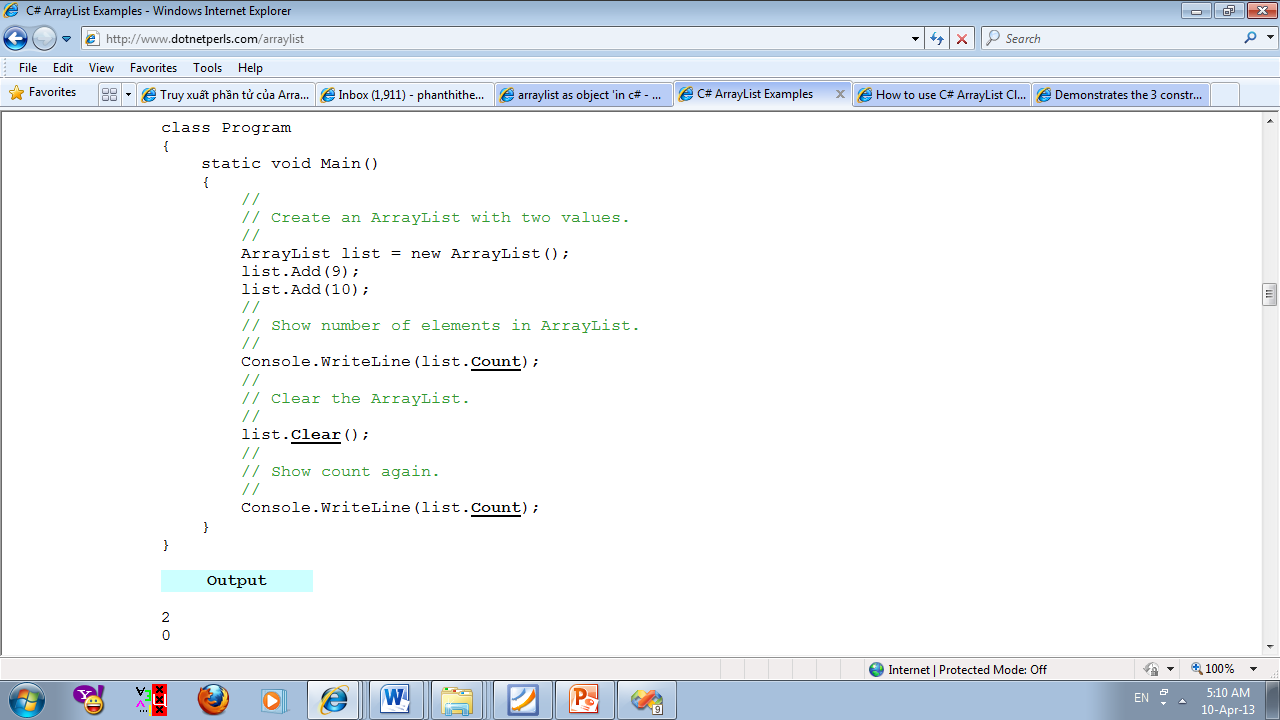
}

1. **ARRAYLIST**

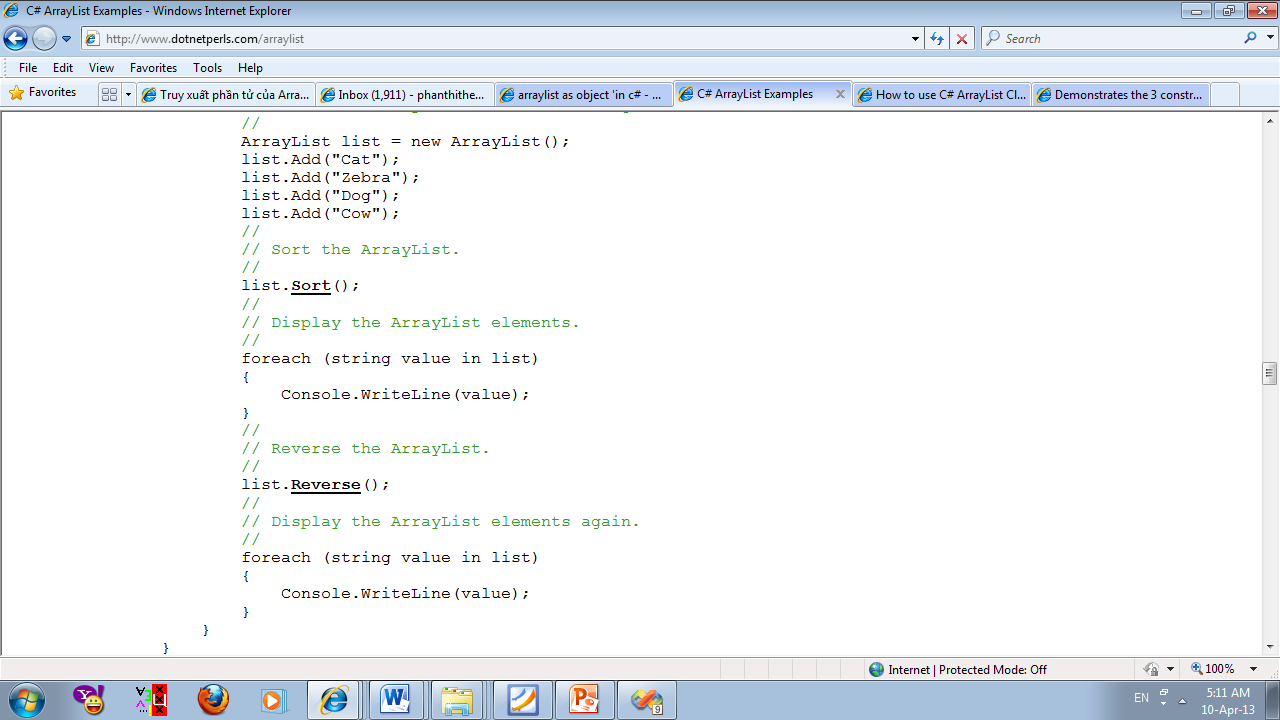
Bài 1: Sử dụng hàm AddRange, Kết hợp 2 mảng



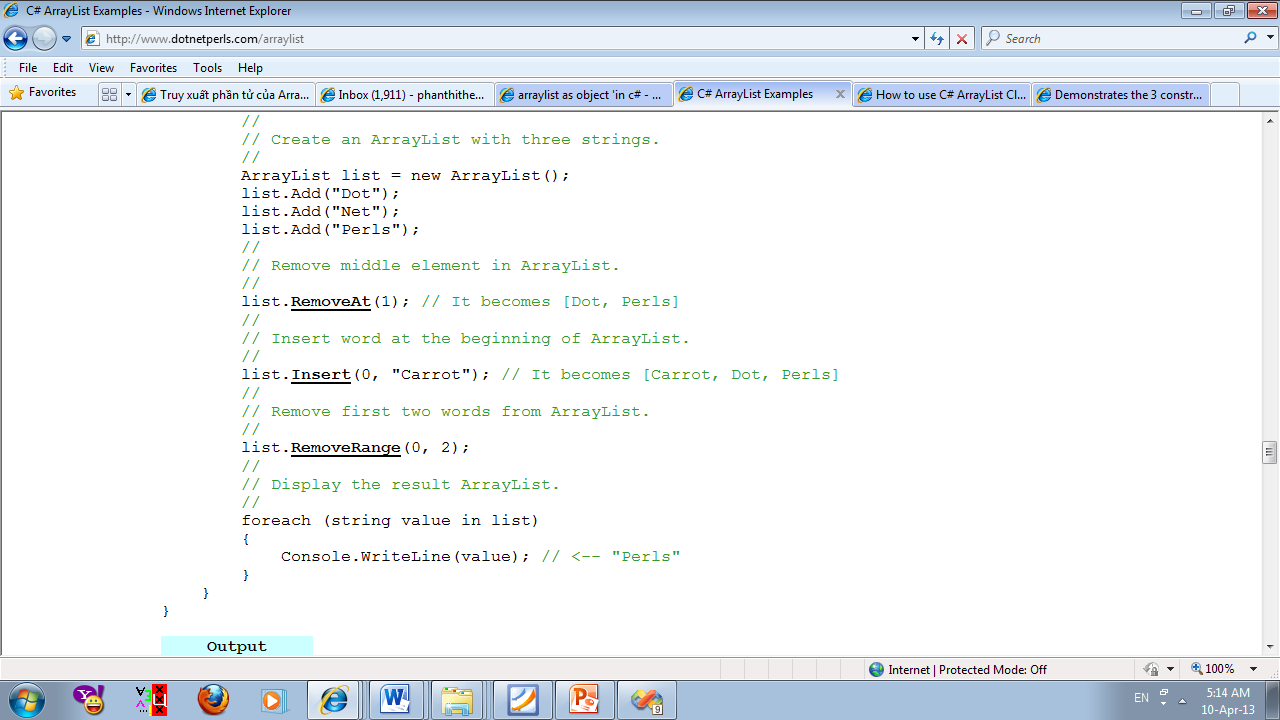
Bài 2: Sử dụng hàm Count, Clear



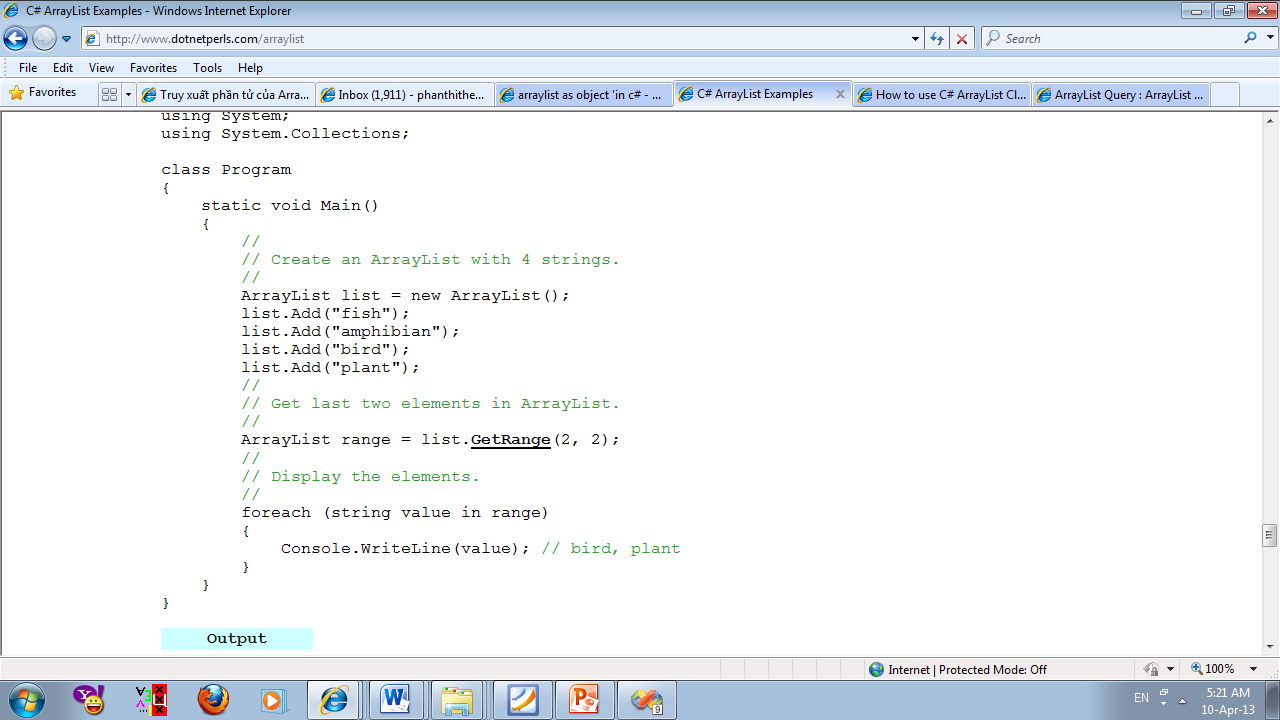
Bài 3: Sử dụng hàm Sort và Reverse



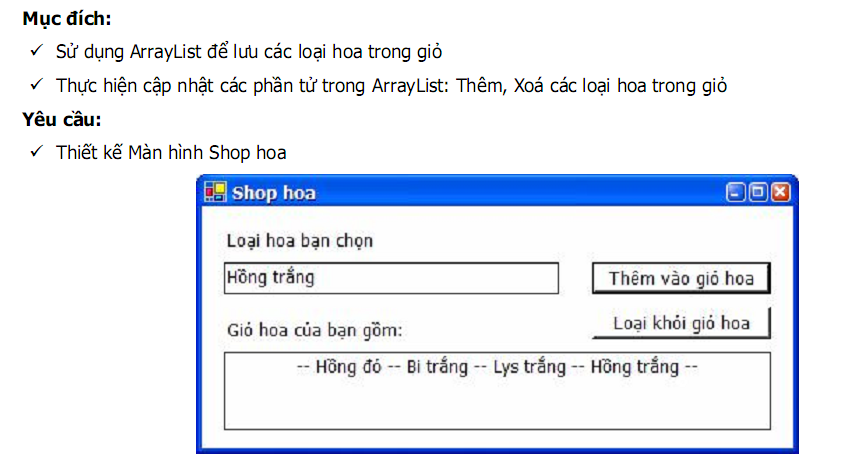
Bài 4: Sử dụng hàm Insert, Remove



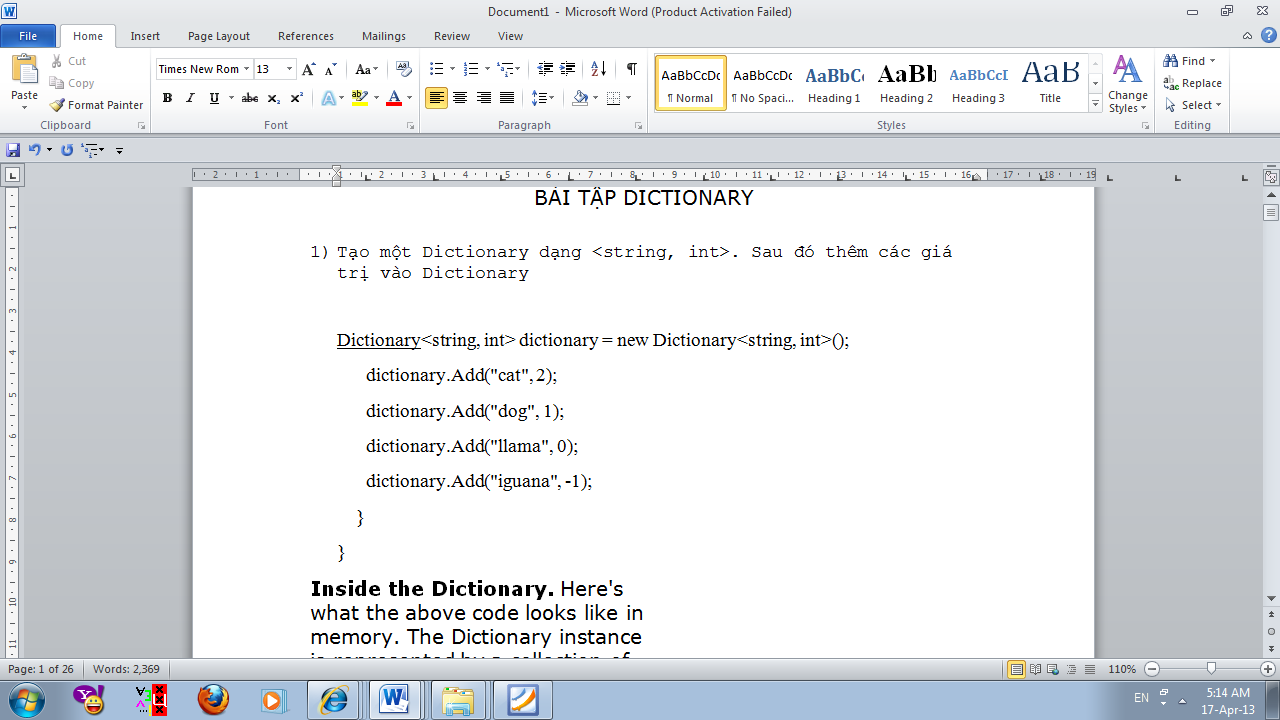
Bài 5: Sử dụng hàm GetRange



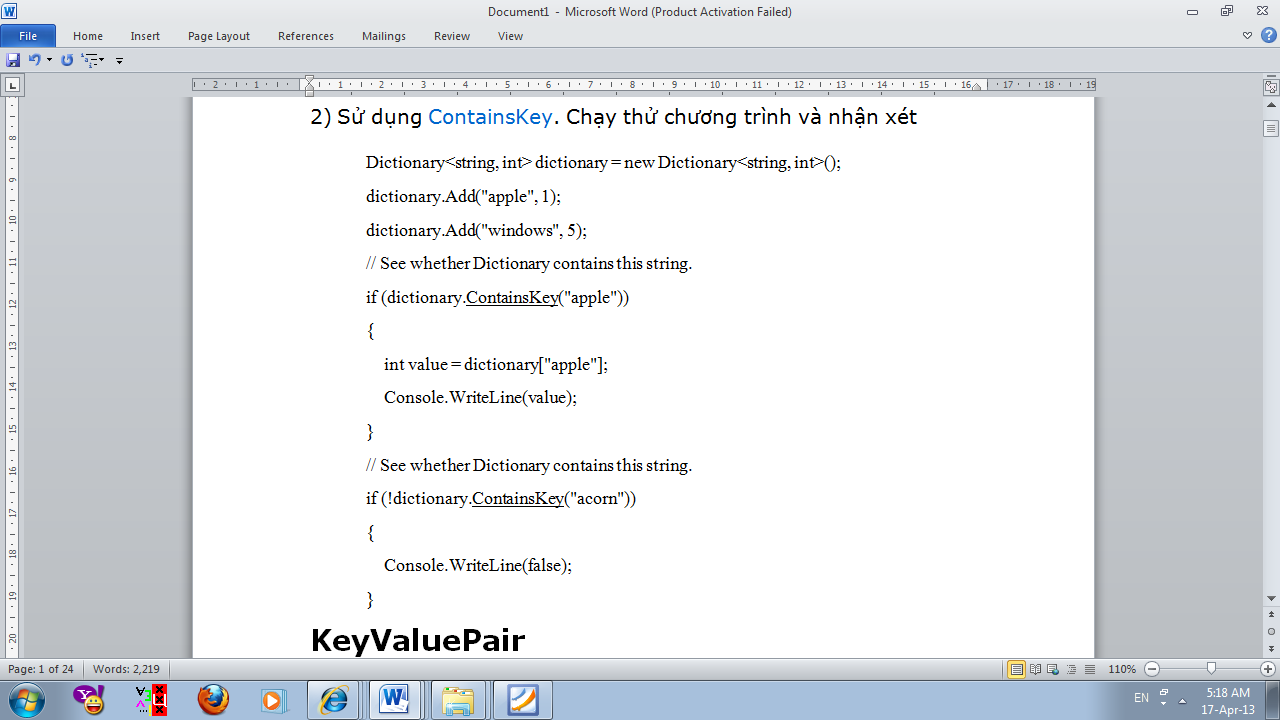
Bài 6: Thiết kế Shop Hoa như sau:



1. **DICTIONARY**
2. Tạo một Dictionary dạng <string, int>. Sau đó thêm các giá trị vào Dictionary



1. Sử dụng [ContainsKey](http://www.dotnetperls.com/containskey). Chạy thử chương trình và nhận xét



1. **Sử dụng KeyValuePair**

class Program

{

static void Main()

{

// Example Dictionary again

Dictionary<string, int> d = new Dictionary<string, int>()

{

{"cat", 2},

{"dog", 1},

{"llama", 0},

{"iguana", -1}

};

// Loop over pairs with foreach

foreach (KeyValuePair<string, int> pair in d)

{

Console.WriteLine("{0}, {1}",

pair.Key,

pair.Value);

}

// Use var keyword to enumerate dictionary

foreach (var pair in d)

{

Console.WriteLine("{0}, {1}",

pair.Key,

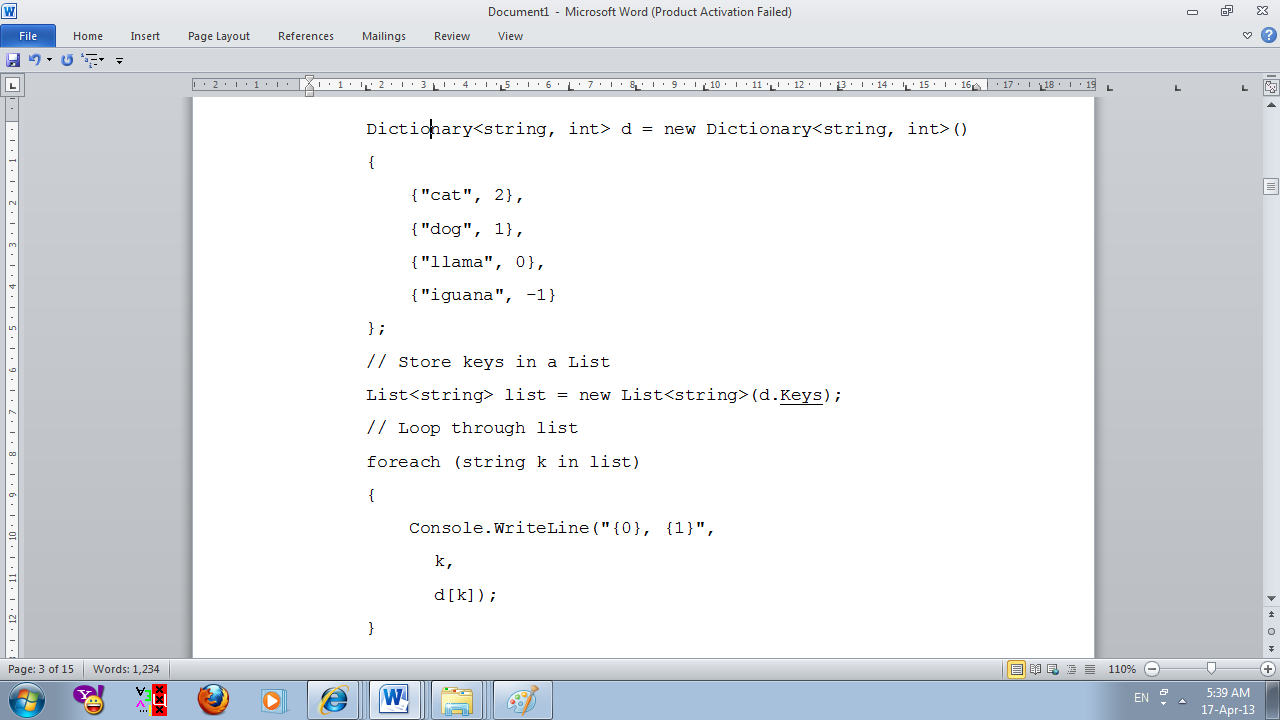
pair.Value);

}

}

}

1. Lấy danh sách các Key đưa vào trong một List



1. Gán giá trị cho Index.

Dictionary<int, int> dictionary = new Dictionary<int, int>();

// You can assign with the indexer.

dictionary[1] = 2;

dictionary[2] = 1;

dictionary[1] = 3; // Reassign.

// You can read with the indexer.

// ... If you read an element that doesn't exist, you get an exception.

Console.WriteLine(dictionary[1]);

Console.WriteLine(dictionary[2]);

}

}

1. Xóa giá trị của Dictionary

Dictionary<string, int> d = new Dictionary<string, int>();

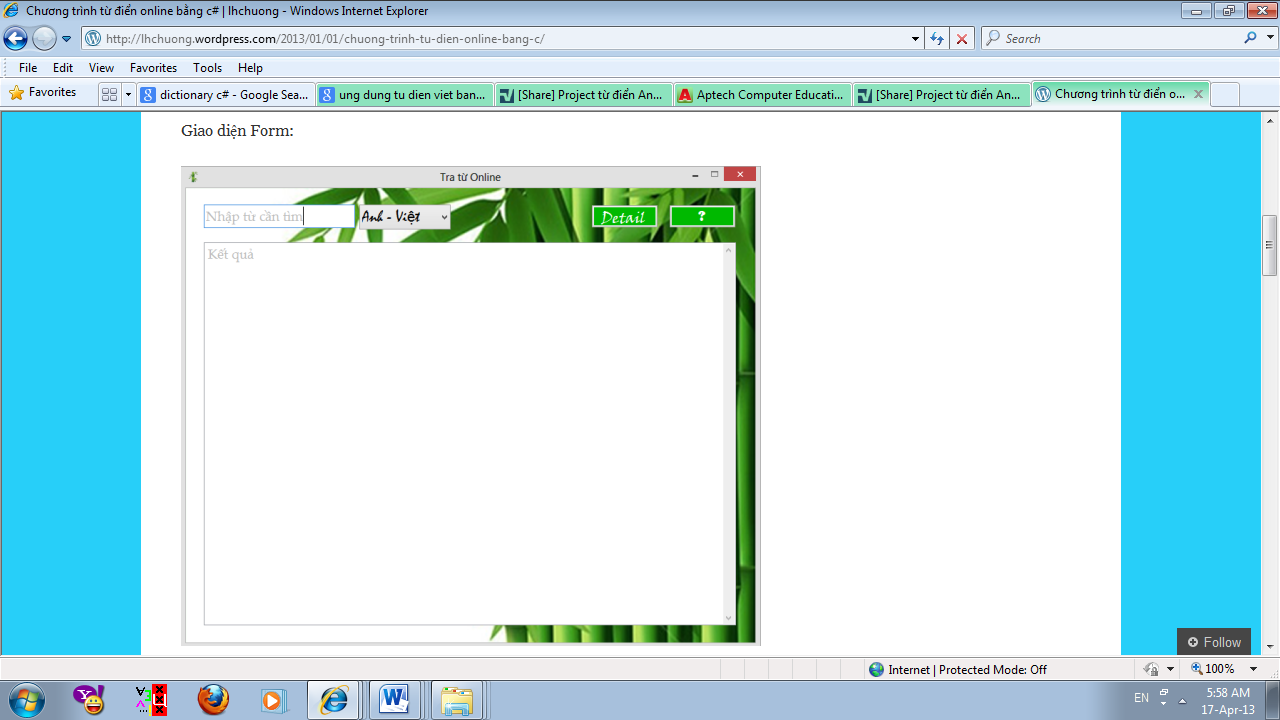
d.Add("cat", 1);

d.Add("dog", 2);

d.Remove("cat"); // Removes cat

d.Remove("nothing"); // Doesn't remove anything

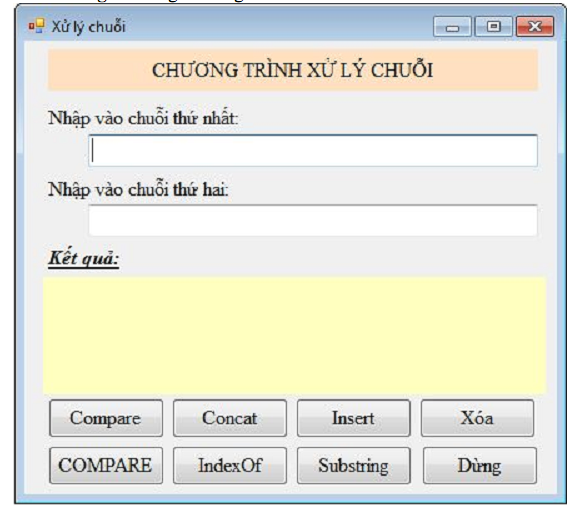
7) Xây dựng một ứng dụng từ điển



1. **STRING**

**BÀI TẬP XỬ LÝ CHUỖI**

Thiết kế form gồm: lblTieuDe, lbl1, txtS1, lbl2, txtS2, lbl3, lblKQ, và các button (xem   
hình).



* Nhắp vào button Compare: so sánh 2 chuỗi txtS1 và txtS2 (có phân biệt chữ HOA và  
  chữ thường), kết quả xuất trong lblKQ.
* Nhắp vào button COMPARE: so sánh 2 chuỗi txtS1 và txtS2 (không phân biệt chữ  
  HOA và chữ thường), kết quả xuất trong lblKQ.
* Nhắp vào button Concat, nối 2 chuỗi txtS1 và txtS2, kết quả xuất trong lblKQ.
* Nhắp button IndexOf, cho biết vị trí xuất hiện của chuỗi txtS2 trong chuỗi txtS1. Nếu  
  có txtS2 trong txtS1 thì thay thế txtS2 (trong txtS1) bằng chuỗi "CHỖ NÀY". Xuất kết quả  
  trong lblKQ.
* Nhắp vào button Insert, chèn chuỗi txtS2 vào sau từ đầu tiên của chuỗi txtS1; và chèn  
  chuỗi txtS2 vào trước từ sau cùng của chuỗi txtS1. Xuất kết quả trong lblKQ.
* Nhắp vào button Substring, cho biết vị trí xuất hiện của chuỗi "TRÌNH XỬ LÝ" trong  
  lblTieuDe. Nếu có thì xóa chuỗi ra khỏi lblTieuDe. Xuất kết quả trong lblKQ.
* Nhắp button Xóa thì xóa trống: TextBox txtS1, TextBox txtS2, Label lblKQ đồng thời  
  đưa con trỏ vào TextBox txtS1.
* Nhắp button Dừng thì dừng chương trình

**G: File và Folder**

**BÀI TẬP THAO TÁC VỚI FILE VÀ FORDER**

1. Chạy đoạn code sau và cho biết kết quả

using System;

using System.IO;

public class FileInformation {

private static void Main(string[] args) {

if (args.Length == 0) {

Console.WriteLine("Please supply a file name:”);

return;

}

FileInfo file = new FileInfo(args[0]);

// Hiển thị thông tin file.

Console.WriteLine("Checking file: " + file.Name);

Console.WriteLine("File exists: " + file.Exists.ToString());

if (file.Exists) {

Console.Write("File created: ");

Console.WriteLine(file.CreationTime.ToString());

Console.Write("File last updated: ");

Console.WriteLine(file.LastWriteTime.ToString());

Console.Write("File last accessed: ");

Console.WriteLine(file.LastAccessTime.ToString());

Console.Write("File size (bytes): ");

Console.WriteLine(file.Length.ToString());

Console.Write("File attribute list: ");

Console.WriteLine(file.Attributes.ToString());

}

Console.WriteLine();

// Hiển thị thông tin thư mục.

DirectoryInfo dir = file.Directory;

Console.WriteLine("Checking directory: " + dir.Name);

Console.WriteLine("In directory: " + dir.Parent.Name);

Console.Write("Directory exists: ");

Console.WriteLine(dir.Exists.ToString());

if (dir.Exists) {

Console.Write("Directory created: ");

Console.WriteLine(dir.CreationTime.ToString());

Console.Write("Directory last updated: ");

Console.WriteLine(dir.LastWriteTime.ToString());

Console.Write("Directory last accessed: ");

Console.WriteLine(dir.LastAccessTime.ToString());

Console.Write("Directory attribute list: ");

Console.WriteLine(dir.Attributes.ToString());

Console.WriteLine("Directory contains: " +

dir.GetFiles().Length.ToString() + " files");

}

Console.ReadLine();

}

}

2) Viết chương trình sao chép một tệp tin từ tệp này sang tệp khác. Tên của tệp được nhập từ bàn phím nguồn và đích được nhập từ tham số dòng lệnh. Ví dụ: cp d:/temp.txt c:/abc.txt