**Mục tiêu**

Luyện tập sử dụng thuật toán đệ quy.

**Mô tả**

Trong phần này, chúng ta sẽ phát triển một ứng dụng cho phép liệt kê cây các thư mục con và file của một thư mục cho trước.

Ứng dụng cho phép nhập vào đường dẫn của một thư mục, sau đó sẽ hiển thị cây bao gồm các thư mục con, cháu… và file ở bên trong thư mục đó

**Hướng dẫn**

Đầu tiên chúng ta xác định yêu cầu bài toán:

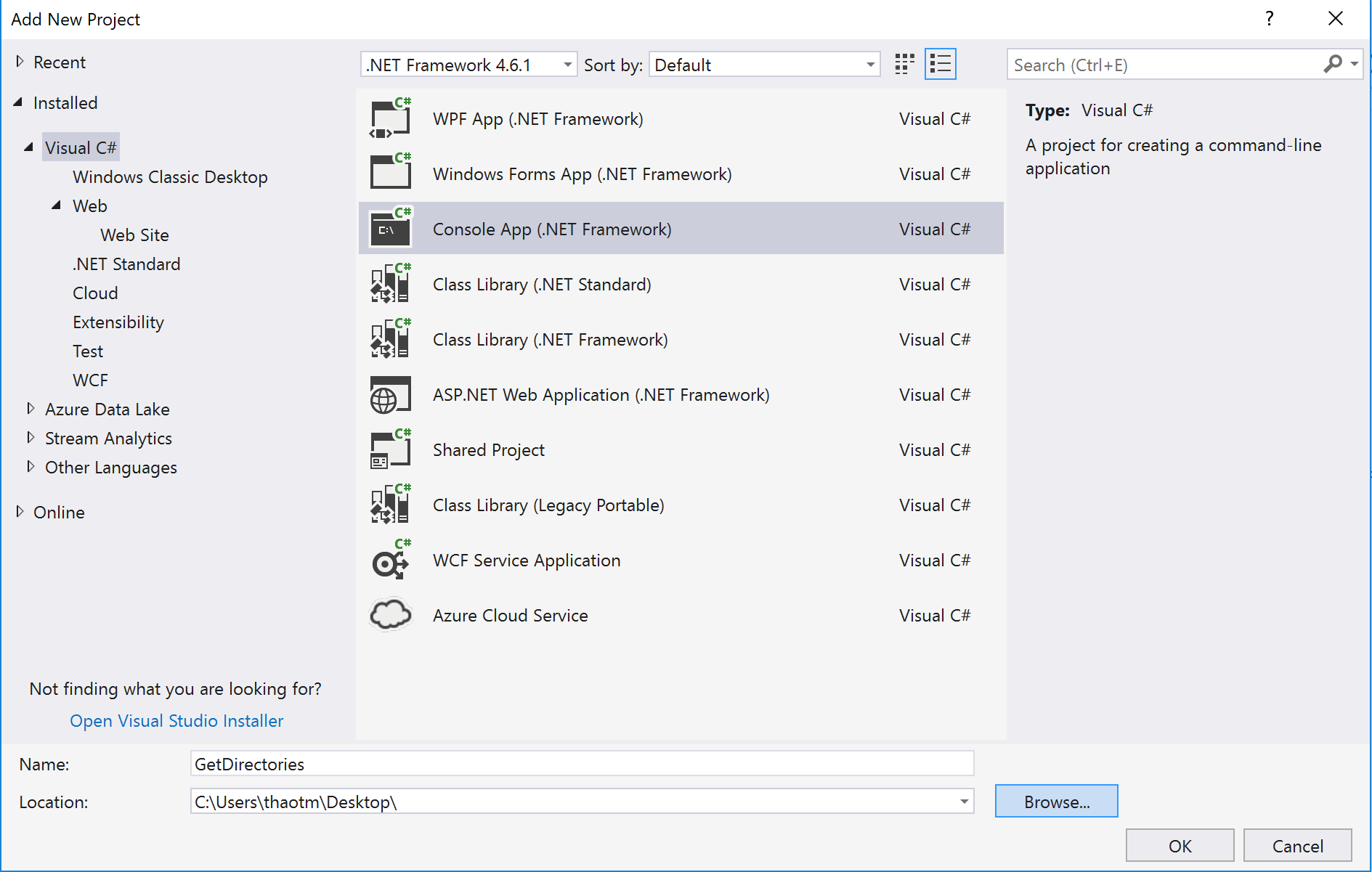
1. Nhâp vào đường dẫn thư mục
2. Tìm các file trong thư mục vừa nhập
3. Tìm tiếp các folder con và các file trong folder con nếu có
4. Tiếp tục như vậy cho đến hết

Như vậy chúng ta cần xử lý bài toán theo các bước như trên. Để làm các bước này chúng ta cần viết 4 hàm:

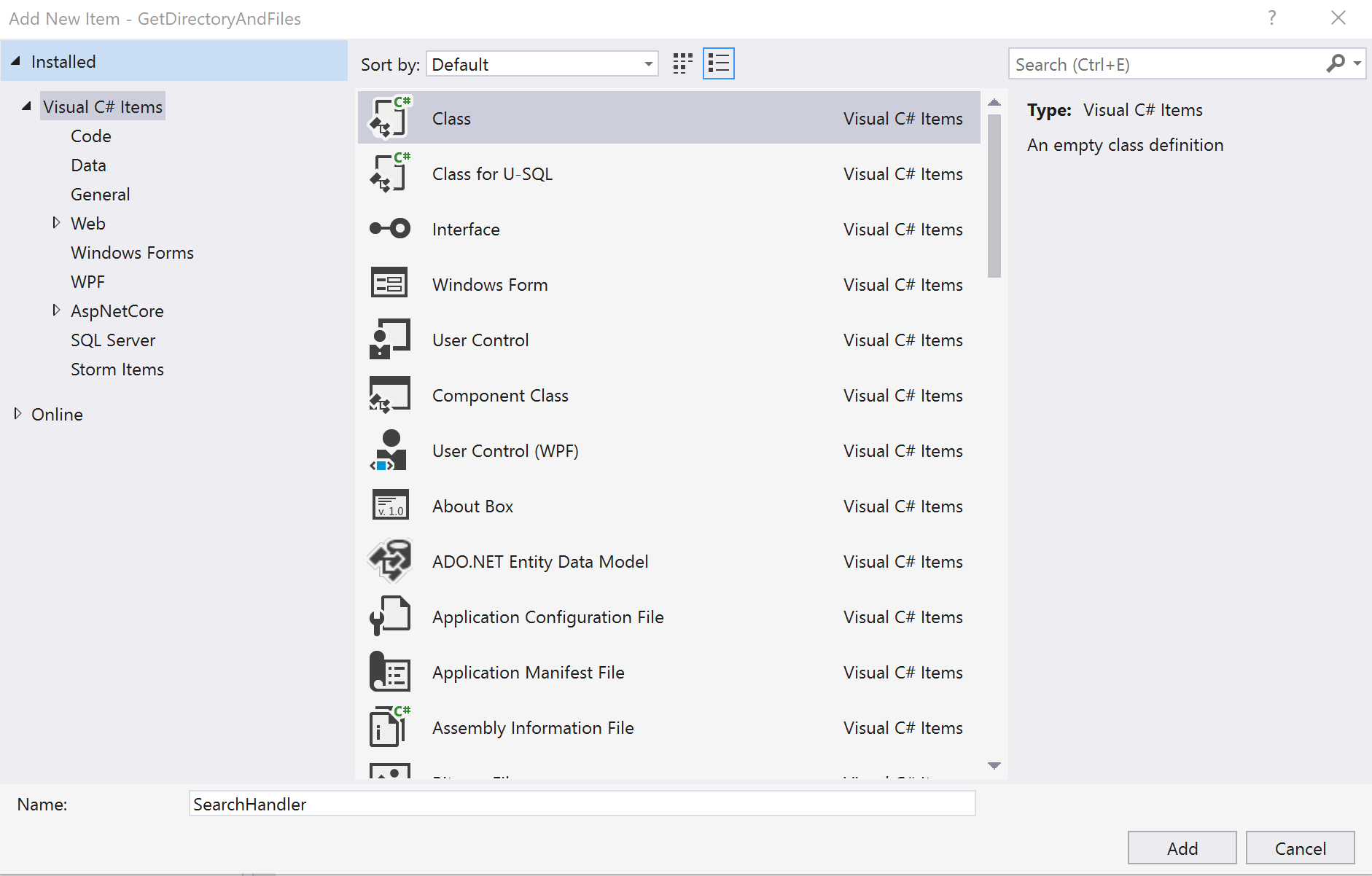
* Hàm đọc đường dẫn folder nhập từ bàn phím (1)
* Hàm đọc danh sách files trong thư mục (2)
* Hàm đọc folder con trong thư mục (3)
* Hàm đệ quy tiếp tục làm việc (2) và (3) cho tới khi hết dữ liệu

Chi tiết như sau:

Tạo mới 1 project **Console Application**, đặt tên là **GetDicrectories,** và chọn nơi lưu project tùy thích, ở đây mình lưu tại desktop



Tiếp theo tạo 1 class đặt tên là **SearchHanlder.cs**, lớp này sẽ chuyên xử lý các công việc tìm kiếm files, folders.



Màn hình tạo mới lớp SearchHandler

Chúng ta thêm 1 hàm tìm kiếm tất cả các file có trong folder như sau:

**public static List<string> GetFiles(string path, string pattern)**

**{**

**try**

**{**

**var listFiles = new List<string>();**

**DirectoryInfo directory = new DirectoryInfo(path);**

**FileInfo[] Files = directory.GetFiles(pattern);**

**string str = string.Empty;**

**foreach (FileInfo file in Files)**

**{**

**listFiles.Add(file.Name);**

**}**

**return listFiles;**

**}**

**catch (UnauthorizedAccessException)**

**{**

**return new List<string>();**

**}**

**}**

Lưu ý khi làm việc với file các bạn cần thêm namespace System.IO để làm việc

using System;

using System.Collections.Generic;

using System.IO;

Giải thích đoạn code trên:

* Ở đây các bạn có hàm **GetFiles** với 2 tham số là **path** và **pattern.** Trong đó tham số path là đường dẫn của thư mục mà bạn muốn đọc tất cả các file, pattern là loại file mà bạn muốn đọc. Ví dụ nếu muốn đọc file .txt thì các bạn truyền vào pattern = “\*.txt”, nếu muốn lấy hết các loại file thì các bạn truyền pattern = “\*”
* Tiếp theo tạo ra 1 List<string> dùng để lưu trữ tên các file lấy được sau khi tìm kiếm trong folder (với “đường dẫn” là **path**) được truyền ở trên bằng câu lệnh **var listFiles = new List<string>();**
* Đọc tất cả các files trong thư mục

**DirectoryInfo directory = new DirectoryInfo(path)**

**FileInfo[] Files = directory.GetFiles(pattern);**

* Ok giờ tất cả các file đã được lưu trữ trong mảng **Files** ở trên. Việc tiếp theo là loop qua mảng này, lấy ra tên file và thêm chúng vào listFiles rồi return hàm thôi **:)**

**foreach (FileInfo file in Files)**

**{**

**listFiles.Add(file.Name);**

**}**

**return listFiles;**

* Trong hàm này mình dùng cập lệnh **try cacth** đễ xử lý các trường hợp ngoại lệ, nếu trong quá trình đọc file xảy ra lỗi, đoạn lệnh trong catch xảy ra mà mình return về một danh sách files rỗng (không chứa files nào). Nếu việc đọc file diễn ra bình thường hàm sẽ trả về danh sách các file name mà sau này chúng ta sẽ dùng để hiển thị.

Phew mọi thứ có vẻ tương đối ổn, giờ file SearchHandler.cs của chúng ta sẽ trông như sau

**using System;**

**using System.Collections.Generic;**

**using System.IO;**

**using System.Linq;**

**using System.Text;**

**using System.Threading.Tasks;**

**namespace GetDirectoryAndFiles**

**{**

**class SearchHandler**

**{**

**public static List<string> GetFiles(string path, string pattern)**

**{**

**try**

**{**

**var listFiles = new List<string>();**

**DirectoryInfo directory = new DirectoryInfo(path);**

**FileInfo[] Files = directory.GetFiles(pattern);**

**foreach (FileInfo file in Files)**

**{**

**listFiles.Add(file.Name);**

**}**

**return listFiles;**

**}**

**catch (UnauthorizedAccessException)**

**{**

**return new List<string>();**

**}**

**}**

**}**

**}**

Vậy là chúng ta đã xử lý xong bước (2) của yêu cầu rồi nhé. Giờ đến bước (3) nào.

Tiếp tục với lớp SearchHandler.cs, chúng ta sẽ thêm hàm lấy đường dẫn các folder con trong một đường dẫn bằng cách viết thêm hàm **GetDirectories**  trông như sau:

**public static List<string> GetDirectories(string path)**

**{**

**try**

**{**

**return Directory.GetDirectories(path).ToList();**

**}**

**catch (UnauthorizedAccessException)**

**{**

**return new List<string>();**

**}**

**}**

* Hàm này trả về **danh sách các đường dẫn** tới folder con trong folder cha (được truyền qua biến path) bằng câu lệnh **return Directory.GetDirectories(path).ToList();**
* Nếu có Exception thì trả về danh sách rỗng bằng lệnh **return new List<string>();**

Vậy là xong bước (2) và (3), cũng khá đơn giản phải không nào. Giờ code trong file SearchHandler.cs sẽ trông như sau:

**using System;**

**using System.Collections.Generic;**

**using System.IO;**

**using System.Linq;**

**using System.Text;**

**using System.Threading.Tasks;**

**namespace GetDirectoryAndFiles**

**{**

**class SearchHandler**

**{**

**public static List<string> GetFiles(string path, string pattern)**

**{**

**try**

**{**

**var listFiles = new List<string>();**

**DirectoryInfo directory = new DirectoryInfo(path);**

**FileInfo[] Files = directory.GetFiles(pattern);**

**foreach (FileInfo file in Files)**

**{**

**listFiles.Add(file.Name);**

**}**

**return listFiles;**

**}**

**catch (UnauthorizedAccessException)**

**{**

**return new List<string>();**

**}**

**}**

**public static List<string> GetDirectories(string path)**

**{**

**try**

**{**

**return Directory.GetDirectories(path).ToList();**

**}**

**catch (UnauthorizedAccessException)**

**{**

**return new List<string>();**

**}**

**}**

**}**

**}**

Vậy là chúng ta đã đi được 50% quãng đường rồi. Tiếp theo chúng ta sẽ xử lý bước (4) của bài toán.

* Thêm 2 hàm **GetHierarchicalFolderList** và **GetHierarchicalChildItems** trong file **Program.cs**
* Ý tưởng ở đây là: hàm **GetHierarchicalFolderList** sẽ làm nhiệm vụ đọc ra các folder con trong đường dẫn mà chúng ta nhập vào, sau đó với mỗi folder con sẽ đọc ra các files hiện có bằng cách gọi hàm **GetHierarchicalChildItems**

**static void GetHierarchicalFolderList(string path, string searchPattern = "\*")**

**{**

**var folders = SearchHandler.GetDirectories(path);**

**string folder = string.Empty;**

**foreach (var item in folders)**

**{**

**Console.WriteLine(item);**

**folder = item;**

**GetHierarchicalChildItems(ref folder, searchPattern, 1);**

**}**

**}**

**static void GetHierarchicalChildItems(ref string folder, string searchPattern, int level)**

**{**

**var files = SearchHandler.GetFiles(folder, searchPattern);**

**if (files.Count <= 0) return;**

**var indent = string.Empty;**

**var splash = string.Empty;**

**for (var i = 0; i < level; i++)**

**{**

**indent += "---";**

**splash += "| ";**

**}**

**foreach (var file in files)**

**Console.WriteLine(splash + indent + file);**

**var folders = SearchHandler.GetDirectories(folder);**

**if (folders.Count > 0)**

**{**

**string subFolder = string.Empty;**

**foreach (var item in folders)**

**{**

**Console.WriteLine(splash + item);**

**subFolder = item;**

**GetHierarchicalChildItems(ref subFolder, searchPattern, level + 1);**

**}**

**}**

**}**

Phân tích một chút nào:

* Hàm **GetHierarchicalFolderList** lấy danh sách các folder có trong đường dẫn path qua tham số mà chúng ta truyền vào bằng câu lệnh **var folders = SearchHandler.GetDirectories(path);** Ở đây chúng ta sử dụng hàm **GetDirectories** của lớp **SearchHandler** đã viết ở trên.
* Tiếp theo hàm này loop qua các folder con (nếu có) và thực hiện 2 việc:
  + Ghi ra đường dẫn tới folder con
  + Đọc tất cả các file trong folder con bằng cách gọi hàm **GetHierarchicalChildItems**
* **GetHierarchicalChildItems** sẽ làm 2 việc:
  + Đọc tất cả các files trong folder
  + Tìm xem folder hiện tại có chứa folder con không, nếu có tự gọi chính nó 1 lần nữa (thuật toán đệ quy). Việc này sẽ tiến hành tới thư mục con cuối cùng.

Phew, xong bước (4) rồi. Giờ code trong file Program.cs của chúng ta sẽ trông như sau:

**using System;**

**using System.Collections.Generic;**

**using System.IO;**

**using System.Linq;**

**using System.Text;**

**using System.Threading.Tasks;**

**namespace GetDirectoryAndFiles**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**}**

**static void GetHierarchicalFolderList(string path, string searchPattern = "\*")**

**{**

**var folders = SearchHandler.GetDirectories(path);**

**string folder = string.Empty;**

**foreach (var item in folders)**

**{**

**Console.WriteLine(item);**

**folder = item;**

**GetHierarchicalChildItems(ref folder, searchPattern, 1);**

**}**

**}**

**static void GetHierarchicalChildItems(ref string folder, string searchPattern, int level)**

**{**

**var files = SearchHandler.GetFiles(folder, searchPattern);**

**if (files.Count <= 0) return;**

**var indent = string.Empty;**

**var splash = string.Empty;**

**for (var i = 0; i < level; i++)**

**{**

**indent += "---";**

**splash += "| ";**

**}**

**foreach (var file in files)**

**Console.WriteLine(splash + indent + file);**

**var folders = SearchHandler.GetDirectories(folder);**

**if (folders.Count > 0)**

**{**

**string subFolder = string.Empty;**

**foreach (var item in folders)**

**{**

**Console.WriteLine(splash + item);**

**subFolder = item;**

**GetHierarchicalChildItems(ref subFolder, searchPattern, level + 1);**

**}**

**}**

**}**

**}**

**}**

Nãy giờ chúng ta đã đi được ¾ quãng đường. Giờ là lúc hoàn thành nốt bài tập này bằng việc xử lý nốt bước 1.

Trong hàm Main của file Program.cs chúng ta thêm đoạn code sau:

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please input directory: ");**

**var directory = Console.ReadLine();**

**Console.WriteLine("\nResult:");**

**if (Directory.Exists(directory))**

**GetHierarchicalFolderList(directory, "\*");**

**else**

**Console.WriteLine("Not Exist!");**

**Console.ReadLine();**

**}**

Rồi đó, trông khá đơn giản nhỉ. Bằng việc đọc những gì được nhập vào qua lệnh ReadLine() chúng ta kiểm tra xem đường dẫn thư mục vừa nhập vào có tồn tại không bằng lệnh **if (Directory.Exists(directory))** nếu đường dẫn hợp lệ chúng ta tiến hành chạy chương trình bằng việc gọi hàm **GetHierarchicalFolderList(directory, "\*");** nếu không dừng chương trình và hiện thông báo :)

Code của file Program.cs cuối cùng như sau:

**using System;**

**using System.Collections.Generic;**

**using System.IO;**

**using System.Linq;**

**using System.Text;**

**using System.Threading.Tasks;**

**namespace GetDirectoryAndFiles**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Console.WriteLine("Please input directory: ");**

**var directory = Console.ReadLine();**

**Console.WriteLine("\nResult:");**

**if (Directory.Exists(directory))**

**GetHierarchicalFolderList(directory, "\*");**

**else**

**Console.WriteLine("Not Exist!");**

**Console.ReadLine();**

**}**

**static void GetHierarchicalFolderList(string path, string searchPattern = "\*")**

**{**

**var folders = SearchHandler.GetDirectories(path);**

**string folder = string.Empty;**

**foreach (var item in folders)**

**{**

**Console.WriteLine(item);**

**folder = item;**

**GetHierarchicalChildItems(ref folder, searchPattern, 1);**

**}**

**}**

**static void GetHierarchicalChildItems(ref string folder, string searchPattern, int level)**

**{**

**var files = SearchHandler.GetFiles(folder, searchPattern);**

**if (files.Count <= 0) return;**

**var indent = string.Empty;**

**var splash = string.Empty;**

**for (var i = 0; i < level; i++)**

**{**

**indent += "---";**

**splash += "| ";**

**}**

**foreach (var file in files)**

**Console.WriteLine(splash + indent + file);**

**var folders = SearchHandler.GetDirectories(folder);**

**if (folders.Count > 0)**

**{**

**string subFolder = string.Empty;**

**foreach (var item in folders)**

**{**

**Console.WriteLine(splash + item);**

**subFolder = item;**

**GetHierarchicalChildItems(ref subFolder, searchPattern, level + 1);**

**}**

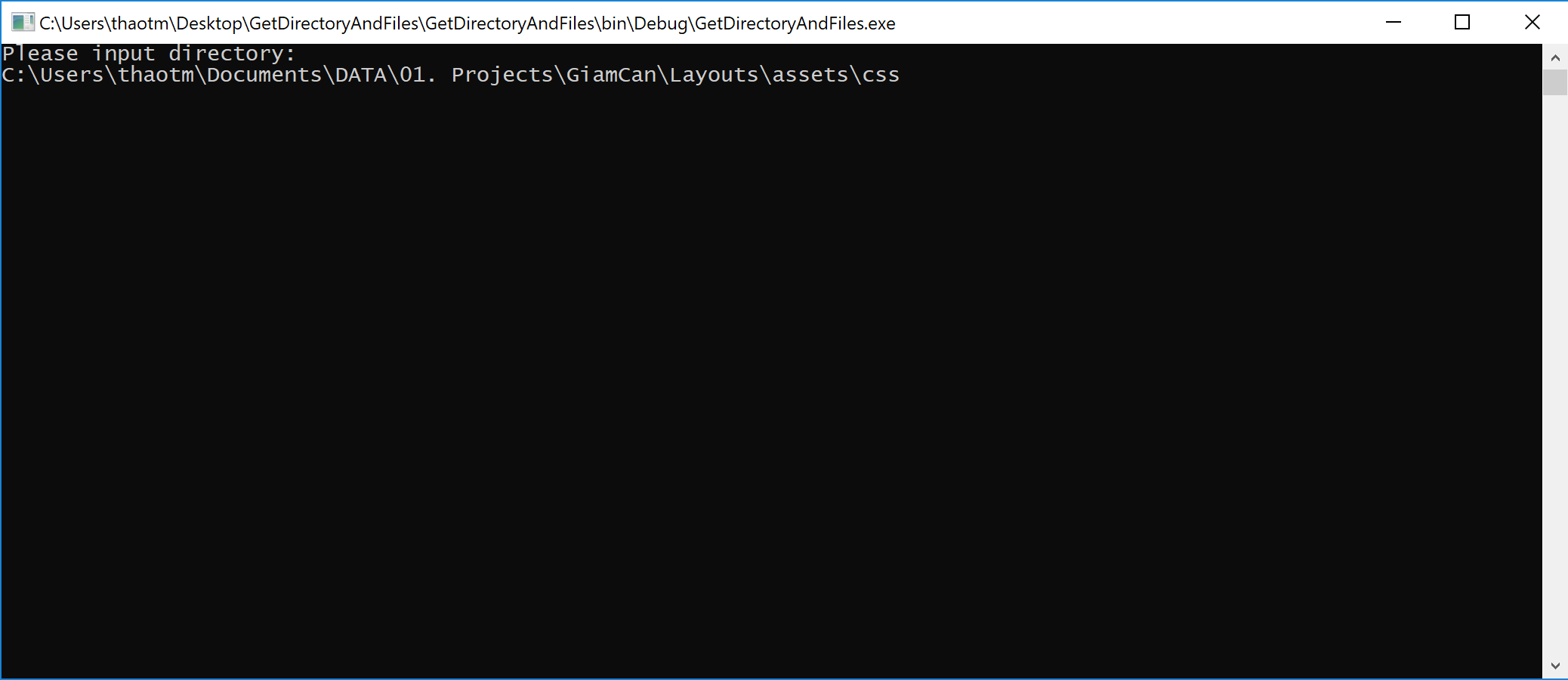
**}**

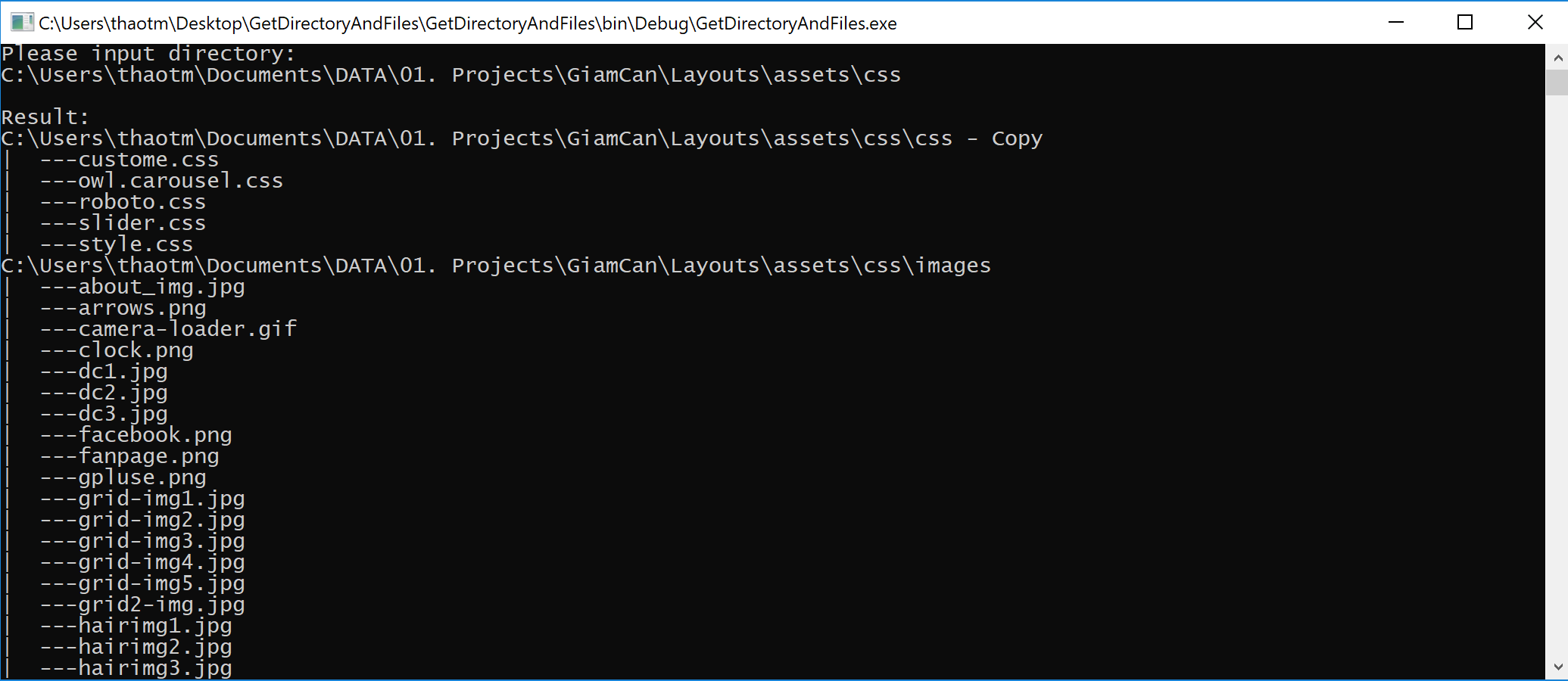
**}**

**}**

**}**

Bước cuối cùng, chạy thử chương trình và xem kết quả.





**Tóm tắt**: Ở bài này các bạn đã thực hành việc sử dụng thuật toán đệ quy để làm việc với File và thư mục qua lớp System.IO.

Chúc các bạn học tốt.