

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

---==🖎🕮🖎==---



**BÀI TẬP LỚN: HỆ CHUYÊN GIA**

**Đề tài:** Tìm hiểu và xây dựng

hệ chuyên gia tư vấn ngôn ngữ lập trình

**Giáo viên hướng dẫn:**

**Nhóm:**

**Thành viên:**

1. Tạ Minh Luận

2. Hoàng Nghĩa Mạnh

3. Nguyễn Thành Đạt

4. Nguyễn Văn Thắng

 5. Phạm Minh Phương

Bài tập lớn hệ chuyên gia

[CHƯƠNG I: GIỚI THIỆU VỀ HỆ CHUYÊN GIA 3](#_Toc481416087)

[1.1 Hệ chuyên gia 3](#_Toc481416088)

[1.2 Các thành phần của hệ chuyên gia 3](#_Toc481416089)

[1.3 Đặc trưng và ưu điểm của hệ chuyên gia 4](#_Toc481416090)

[1.3.1 Có bốn đặc trưng cơ bản của một hệ chuyên gia : 4](#_Toc481416091)

[1.3.2 Những ưu điểm của hệ chuyên gia : 4](#_Toc481416092)

[CHƯƠNG II: XÂY DỰNG HỆ CHUYÊN GIA TRONG PROLOG 6](#_Toc481416093)

[2.1 Giới thiệu về prolog 6](#_Toc481416094)

[2.1.1 Prolog là ngôn ngữ lập trình logic 6](#_Toc481416095)

[2.1.2 Cú pháp của prolog 6](#_Toc481416096)

[2.2 Kết nối prolog với C# 7](#_Toc481416097)

[2.1.1 Prolog và C# 7](#_Toc481416098)

[2.1.2 Các hàm kết nối và truy vấn 8](#_Toc481416099)

[CHƯƠNG III: XÂY DỰNG ỨNG DỤNG 10](#_Toc481416100)

[3.1 Thống kê 10](#_Toc481416101)

[3.2 Xây dựng CSTT 10](#_Toc481416102)

[3.2.1 Sự kiện 10](#_Toc481416103)

[3.2.2 Các tập luật 11](#_Toc481416104)

[3.2.3 File prolog 12](#_Toc481416105)

[3.3 Hệ chuyên gia tư vấn ngôn ngữ lập trình 13](#_Toc481416106)

# CHƯƠNG I: GIỚI THIỆU VỀ HỆ CHUYÊN GIA

## 1.1 Hệ chuyên gia

Theo E.Feigenbaum : «Hệ chuyên gia (Expert System) là một chương trình máy tính thông minh sử dụng tri thức (knowledge) và các thủ tục suy luận (inference procedures) để giải những bài toán tương đối khó khăn đòi hỏi những chuyên gia mới giải được».

Hệ chuyên gia là một hệ thống tin học có thể mô phỏng (emulates) năng lực quyết đoán (decision) và hành động (making abilily) của một chuyên gia (con người). Hệ chuyên gia là một trong những lĩnh vực ứng dụng của trí tuệ nhân tạo (Artificial Intelligence) như hình dưới đây.

Hệ chuyên gia sử dụng các tri thức của những chuyên gia để giải quyết các vấn đề (bài toán) khác nhau thuộc mọi lĩnh vực.

Tri thức (knowledge) trong hệ chuyên gia phản ánh sự tinh thông được tích tụ từ sách vở,tạp chí, từ các chuyên gia hay các nhà bác học. Các thuật ngữ hệ chuyên gia, hệ thống dựa trên tri thức (knowledge- based system )hay hệ chuyên gia dựa trên tri thức(knowledge- based expert system) thường có cùng nghĩa.

## 1.2 Các thành phần của hệ chuyên gia

Một hệ chuyên gia gồm ba thành phần chính là

cơ sở tri thức (knowledge base),

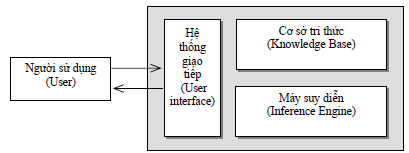
máy suy diễn hay môtơ suy diễn (inference engine),

và hệ thống giao tiếp với người sử dụng (user interface).

Cơ sở tri thức chứa các tri thức để từ đó, máy suy diễn tạo ra câu trả lời cho ngườisử dụng qua hệ thống giao tiếp.

Người sử dụng (user) cung cấp sự kiện (facts) là những gì đã biết, đã có thật hay những thông tin có ích cho hệ chuyên gia, và nhận được những câu trả lời là những lời khuyên hay những gợi ý đúng đắn (expertise).

Hoạt động của một hệ chuyên gia dựa trên tri thức được minh họa như sau :



*Hình 1.1: Hoạt động của hệ chuyên gia*

## 1.3 Đặc trưng và ưu điểm của hệ chuyên gia

### 1.3.1 Có bốn đặc trưng cơ bản của một hệ chuyên gia :

Hiệu quả cao (high performance). Khả năng trả lời với mức độ tinh thông bằng hoặc cao hơn so với chuyên gia (người) trong cùng lĩnh vực.

Thời gian trả lời thoả đáng (adequate response time). Thời gian trả lời hợp lý, bằng hoặc nhanh hơn so với chuyên gia (người) để đi đến cùng một quyết định. Hệ chuyên gia là một hệ thống thời gian thực (real time system).

Độ tin cậy cao (good reliability). Không thể xảy ra sự cố hoặc giảm sút độ tin cậy khisử dụng.

Dễ hiểu (understandable). Hệ chuyên gia giải thích các bước suy luận một cách dễ hiểuvà nhất quán, không giống như cách trả lời bí ẩn của các hộp đen (black box).

### 1.3.2 Những ưu điểm của hệ chuyên gia :

Phổ cập (increased availability).Là sản phẩm chuyên gia, được phát triển không ngừng với hiệu quả sử dụng không thể phủ nhận.

Giảm giá thành (reduced cost).

Giảm rủi ro (reduced dangers). Giúp con người tránh được trong các môi trường rủi ro, nguy hiểm.

Tính thường trực (Permanance).Bất kể lúc nào cũng có thể khai thác sử dụng, trong khi con người có thể mệt mỏi, nghỉ ngơi hay vắng mặt.

Đa lĩnh vực (multiple expertise). chuyên gia về nhiều lĩnh vực khác nhau và được khai thác đồng thời bất kể thời gian sử dụng.

Độ tin cậy (increased relialility). Luôn đảm bảo độ tin cậy khi khai thác.

Khả năng giảng giải (explanation). Câu trả lời với mức độ tinh thông được giảng giải rõ ràng chi tiết, dễ hiểu.

Khả năng trả lời (fast reponse). Trả lời theo thời gian thực, khách quan.

Tính ổn định, suy luận có lý và đầy đủ mọi lúc mọi nơi (steady, une motional, and complete response at all times).

Trợ giúp thông minh như một người hướng dẫn (intelligent -tutor).

Có thể truy cập như là một cơ sở dữ liệu thông minh (intelligent database).

# CHƯƠNG II: XÂY DỰNG HỆ CHUYÊN GIA TRONG PROLOG

## 2.1 Giới thiệu về prolog

### 2.1.1 Prolog là ngôn ngữ lập trình logic

Prolog là ngôn ngữ lập trình được sử dụng và phổ biến nhất trong các ngôn ngữ lập trình logic (cụm từ Prolog có nghĩa là Programming in Logic). Ngôn ngữ này do giáo sư người Pháp Alain Colmetanuer và nhóm nghiên cứu của ông đề xuất trong năm 1970.

Prolog còn được gọi là ngôn ngữ lập trình kí hiệu (symbolic programming), tương tự như các ngôn ngữ lập trình khác nó thích hợp cho việc giải bài toán liên quan đến đối tượng và mối liên hệ giữa chúng…

Prolog được sử dụng phổ biến trong lĩnh vực trí tuệ nhân tạo. Nguyên lý lập trình logic dưa trên mệnh đề Horn. Một mệnh đề Horn biểu diễn một sự kiện hay một sự việc nào đó đúng hay không đúng, xảy ra hay không xảy ra.

Ví dụ : - Nếu Nam là cha mẹ của Bình, Bình là cha mẹ của Bắc thì Nam là ông của Bắc…

### 2.1.2 Cú pháp của prolog

#### 2.1.2.1 Các thuật ngữ

Một chương trình Prolog là một cơ sở dữ liệu gồm các mệnh đề (clause). Mỗi mệnh đề được xây dựng từ các vị từ (predicat). Một đơn vị từ phát biểu về đối tượng có giá trị chân lý đúng (true) hoặc sai (false).

Mỗi nguyên tử biểu diễn một quan hệ giữa các hạng. Hạng và quan hệ giữa các hạng tạo thành mệnh đề.

Hạng là đối tượng dữ liệu có thể là hạng sơ cấp (elementary) gồm hằng (constaint) biến (variable) và các hạng phức hợp (compound term).

Hạng phức hợp là một hàm tử (functor) có chứa các đối (argument) có dạng:

Tên\_hàm\_tử (Đối\_1, ..., Đối\_n)

Tên hàm thường bắt đầu bằng chuỗi chữ cái hoặc chữ số được bắt đầu bằng chữ cái thường.

Trong Prolog hàm tử đặc biệt “.” (dấu chấm) biểu diễn cấu trúc danh sách (list).

Ví dụ : t (a, b, 10).

Student (long, 1996, diachi (‘NgoaLong’, ‘Nhổn’)).

Mệnh đề có thể là một sự kiện, một luật, hay một câu hỏi.

Quy ước viết sau mỗi mệnh đề 1 dấu chấm:

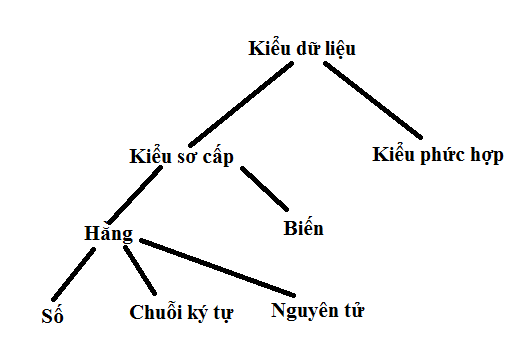
+ Sự kiện: <…>.

(tương ứng với luật <…>: - true.)

+ Luật: <…>: - <…>.

+ Câu hỏi: ? - <…>.

#### 2.1.2.2 Kiểu dữ liệu trong prolog



*Hình 1: Các kiểu dữ liệu trong prolog.*

Cú pháp của Prolog quy định mỗi hiểu đối tượng có một dạng khác nhau. Prolog không cần cung cấp một thông tin nào khác để nhận biết các kiểu của một đối tượng. Trong Prolog, NSD không cần khai báo dữ liệu.

#### 2.1.2.3 Chú thích trong prolog

Chú thích (command) của chương trình Prolog được đặt trong cặp ngoặc /\* chú thich\*/ là chú thích trên một đoạn.

Ví dụ : /\* Hệ chuyên gia tư vấn hướng nghiệp \*/

Chú thích trên một dòng sử dụng kí hiệu %

Ví dụ: % Tim hiểu ngôn ngữ Prolog.

## 2.2 Kết nối prolog với C#

### 2.1.1 Prolog và C#

Prolog là ngôn ngữ lập trình logic. Nó đảm nhận vai trò mô tơ suy diễn

C# là ngôn ngữ mạnh mẽ, phổ biến và dễ học. Nếu như chỉ dùng prolog thì chương trình hệ chuyên gia không có giao diện người dùng.

Trong bài tập lớn chúng tôi sử dụng SWI-Prolog

C#.Net WPF

### 2.1.2 Các hàm kết nối và truy vấn

#### 2.1.2.1 Hàm kết nối

public void Load\_file(string s)

{

s = s.Replace("\\", "//");

s = "consult('" + s + "')";

string query = s.Replace("\\", "//");

try

{

PlQuery q = new PlQuery(query);

Assert.IsTrue(q.NextSolution());

}

catch (SbsSW.SwiPlCs.Exceptions.PlException e)

{

}

}

#### 2.1.2.2 Hàm lấy 1 chuỗi (lấy câu hỏi)

public string Query(string s)

{

s.Trim();

string result = "";

try

{

PlQuery q = new PlQuery(s);

int i = 0;

foreach (PlQueryVariables v in q.SolutionVariables)

{

i++;

if (i >= 10) break;

result += v["X"].ToString() ;

}

return result;

}

catch (SbsSW.SwiPlCs.Exceptions.PlException ex)

{

return "Error query: " + ex.Message;

}

}

#### 2.1.2.3 Hàm lấy nhiều chuỗi (lấy các câu trả lời)

public List<string> GetAnswers(string s)

{

List<string> listResult = new List<string>();

s.Trim();

Regex r = new Regex(@"[A-Z\_][a-zA-Z\_]\*");

MatchCollection matches = r.Matches(s);

try

{

PlQuery q = new PlQuery(s);

int i = 0;

foreach (PlQueryVariables v in q.SolutionVariables)

{

i++;

if (i >= 10) break;

foreach (Match match in matches)

{

string item = v[match.ToString()].ToString();

listResult.Add(item);

}

}

return listResult;

}

catch (SbsSW.SwiPlCs.Exceptions.PlException ex)

{

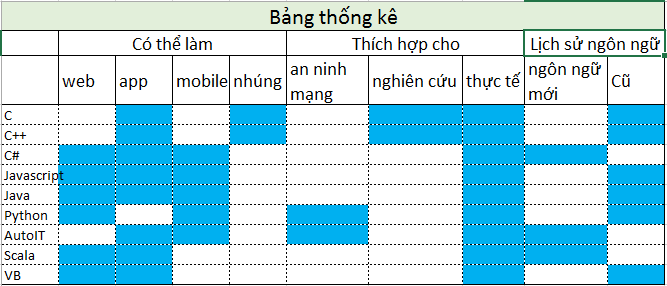
return null;

}

}

# CHƯƠNG III: XÂY DỰNG ỨNG DỤNG

## 3.1 Thống kê



## 3.2 Xây dựng CSTT

### 3.2.1 Sự kiện

#### 3.2.1.1 Các câu hỏi

question(1,'Bạn thích hướng đi nào?').

question(2,'Bạn là người như thế nào?').

question(3,'Bạn muốn một ngôn ngữ mới?').

#### 3.2.1.2 Các lựa chọn

option(1,1.1,'Lập trình web').

option(1,1.2,'Lập trình ứng dụng').

option(1,1.3,'Lập trình di động').

option(1,1.4,'Lập trình nhúng').

option(2,2.1,'Bạn thích an ninh mạng, hacker').

option(2,2.2,'Bạn học KTMT, bạn thích nghiên cứu').

option(2,2.3,'Bạn muốn phần mềm thực tế').

option(3,3.1,'Có').

option(3,3.2,'Không').

#### 3.2.1.3 Các câu trả lời (lời khuyên từ chuyên gia)

answer([1.2,1.4,2.2,2.3,3.2],'C').

answer([1.2,1.4,2.2,2.3,3.2],'C++').

answer([1.1,1.2,1.3,2.3,3.1],'C#').

answer([1.1,1.2,1.3,2.3,3.1,3.2],'Javascript').

answer([1.1,1.2,1.3,2.3,3.1],'Java').

answer([1.1,1.3,2.1,2.3,3.2],'Python').

answer([1.2,1.3,2.1,2.3,3.1],'AutoIT').

answer([1.1,1.2,2.3,3.1],'Scala').

answer([1.1,1.2,2.3,3.2],'VB').

answer([1.1,1.2,1.3,1.4,2.1,2.2,2.3,3.1,3.2],'Javascript').

#### 3.2.1.4 Thông tin các ngôn ngữ

info('C','Ngôn ngữ lập trình C là một ngôn ngữ mệnh lệnh được phát triển từ đầu thập niên 1970 bởi Dennis Ritchie để dùng trong hệ điều hành UNIX. Từ đó, ngôn ngữ này đã lan rộng ra nhiều hệ điều hành khác và trở thành một những ngôn ngữ phổ dụng nhất. C là ngôn ngữ rất có hiệu quả và được ưa chuộng nhất để viết các phần mềm hệ thống, mặc dù nó cũng được dùng cho việc viết các ứng dụng. Ngoài ra, C cũng thường được dùng làm phương tiện giảng dạy trong khoa học máy tính mặc dù ngôn ngữ này không được thiết kế dành cho người nhập môn.').

info('C++','C++ (đọc là "C cộng cộng" hay "xi-plus-plus", IPA: /siː pləs pləs/) là một loại ngôn ngữ lập trình. Đây là một dạng ngôn ngữ đa mẫu hình tự do có kiểu tĩnh và hỗ trợ lập trình thủ tục, dữ liệu trừu trượng, lập trình hướng đối tượng, và lập trình đa hình. Từ thập niên 1990, C++ đã trở thành một trong những ngôn ngữ thương mại phổ biến nhất trong khi đó.').

info('C#','C# (đọc là "C thăng" hay "C sharp" ("xi-sáp")) là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft, là phần khởi đầu cho kế hoạch .NET của họ. Tên của ngôn ngữ bao gồm ký tự thăng theo Microsoft nhưng theo ECMA là C#, chỉ bao gồm dấu số thường. Microsoft phát triển C# dựa trên C++ và Java. C# được miêu tả là ngôn ngữ có được sự cân bằng giữa C++, Visual Basic, Delphi và Java.').

info('Javascript','JavaScript, theo phiên bản hiện hành, là một ngôn ngữ lập trình kịch bản dựa trên đối tượng được phát triển từ các ý niệm nguyên mẫu. Ngôn ngữ này được dùng rộng rãi cho các trang web, nhưng cũng được dùng để tạo khả năng viết script sử dụng các đối tượng nằm sẵn trong các ứng dụng. Nó vốn được phát triển bởi Brendan Eich tại Hãng truyền thông Netscape với cái tên đầu tiên Mocha, rồi sau đó đổi tên thành LiveScript, và cuối cùng thành JavaScript. Giống Java, JavaScript có cú pháp tương tự C, nhưng nó gần với Self hơn Java. .js là phần mở rộng thường được dùng cho tập tin mã nguồn JavaScript.').

info('Java','Java (đọc như "Gia-va") là một ngôn ngữ lập trình hướng đối tượng (OOP) và dựa trên các lớp (class)[9]. Khác với phần lớn ngôn ngữ lập trình thông thường, thay vì biên dịch mã nguồn thành mã máy hoặc thông dịch mã nguồn khi chạy, Java được thiết kế để biên dịch mã nguồn thành bytecode, bytecode sau đó sẽ được môi trường thực thi (runtime environment) chạy.').

info('Python','Python là một ngôn ngữ lập trình thông dịch do Guido van Rossum tạo ra năm 1990. Python hoàn toàn tạo kiểu động và dùng cơ chế cấp phát bộ nhớ tự động; do vậy nó tương tự như Perl, Ruby, Scheme, Smalltalk, và Tcl. Python được phát triển trong một dự án mã mở, do tổ chức phi lợi nhuận Python Software Foundation quản lý.').

info('AutoIT','AutoIt (phát âm aw-toe-it)[2] là một ngôn ngữ lập trình được cung cấp miễn phí, có dạng kịch bản giống như BASIC được thiết kế để tự động hóa các GUI (Graphic User Interface: giao diện người dùng) và các thao tác thường dùng. Nó là sự phối hợp của việc giả lập nhấn phím, di chuyển chuột và thao tác trên cửa sổ/control (control là các đối tượng tương tác được trên cửa sổ, ví dụ như là button - nút bấm) để thực hiện tự động các tác vụ theo cách thức mà các ngôn ngữ khác không thể làm hoặc làm không chuẩn xác. AutoIt có dung lượng rất nhỏ gọn, có thể chạy độc lập trên tất cả các phiên bản của Windows mà không cần các runtime để làm nền. Tuy nhiên, kể từ phiên bản v3.3, AutoIt đã không còn hỗ trợ cho Windows 9x và Win NT 4.0.').

info('Scala','Scala (phát âm /ˈskɑːlə/ SKAH-lə) là một ngôn ngữ lập trình đa mẫu hình, được thiết kế để tích hợp các tính năng của lập trình hướng đối tượng với lập trình hàm[1]. Tên Scala xuất phát từ chữ tiếng Anh scalable, có nghĩa là "ngôn ngữ có khả năng mở rộng", được phát triển dựa trên nhu cầu sử dụng các tính năng mở rộng của nó.').

info('VB','Visual Basic.NET là một công cụ lập trình hỗ trợ ngôn ngữ lập trình Visual Basic.NET trong bộ công cụ lập trình Microsoft Visual Studio 2005. Trong Microsoft Visual Studio 2005, bạn có thể kết hợp các project (dự án) viết bằng các ngôn ngữ khác như C#, J#,... với VB2005. VB2005 được chạy trên nền .NET Framework 2.0, khi đóng gói chương trình cũng cần bộ Framework này để chạy chương trình. Cũng như VB 6.0, VB2005 cũng là ngôn ngữ lập trình hướng đối tượng.').

### 3.2.2 Các tập luật

Người dùng không cần phải chọn tất cả các lựa chọn (đặc tính) mới đưa ra ngôn ngữ đó. Mà chỉ cần các lựa chọn thuộc vào tập thỏa mãn là đưa ra ngôn ngữ đó:

Ví dụ : answer([1.1,1.2,1.3,2.3,3.1,3.2],'Javascript').

Người dùng không phải chọn tất cả các lựa chọn 1.1, 1.2, 1.3, 3.1, 3.2

Mà chỉ cần 1 hoặc vài trong số chúng thì chuyên gia vẫn đưa ra ngôn ngữ Javascript

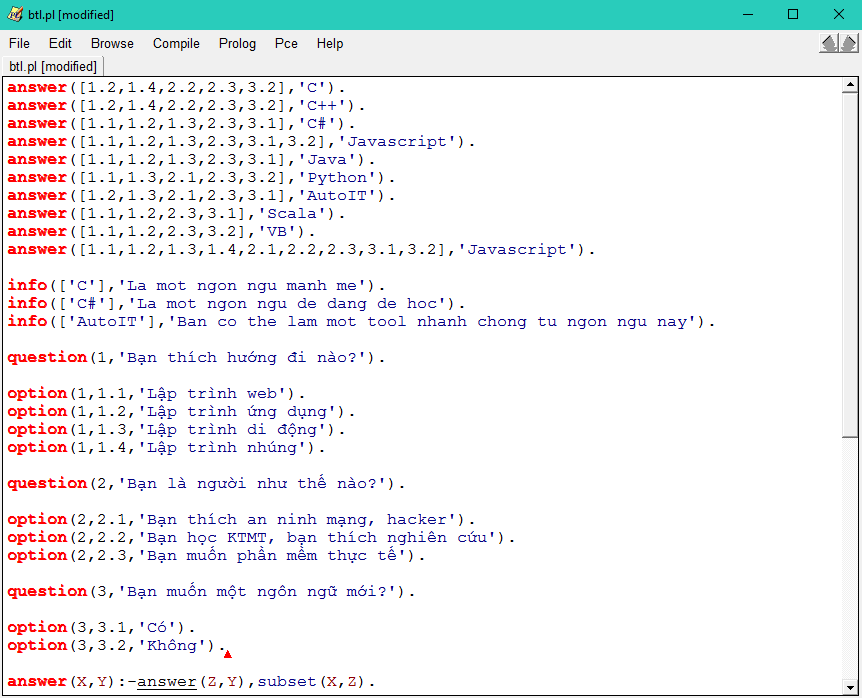
Bởi vì, thứ nhất chuyên gia sẽ đưa ra các ngôn ngữ thỏa mãn được nhu cầu của người dùng. Còn bản thân ngôn ngữ đó có thể có các đặc tính khác. Thứ hai nếu đưa ra ngôn ngữ khớp với các lựa chọn tỉ lệ khá thấp nhiều khi chuyên gia sẽ không đưa ra được ngôn ngữ nào phù hợp

Khi thu thập các lựa chọn X thì sẽ đưa ra các ngôn ngữ Y với X, Y thỏa mãn

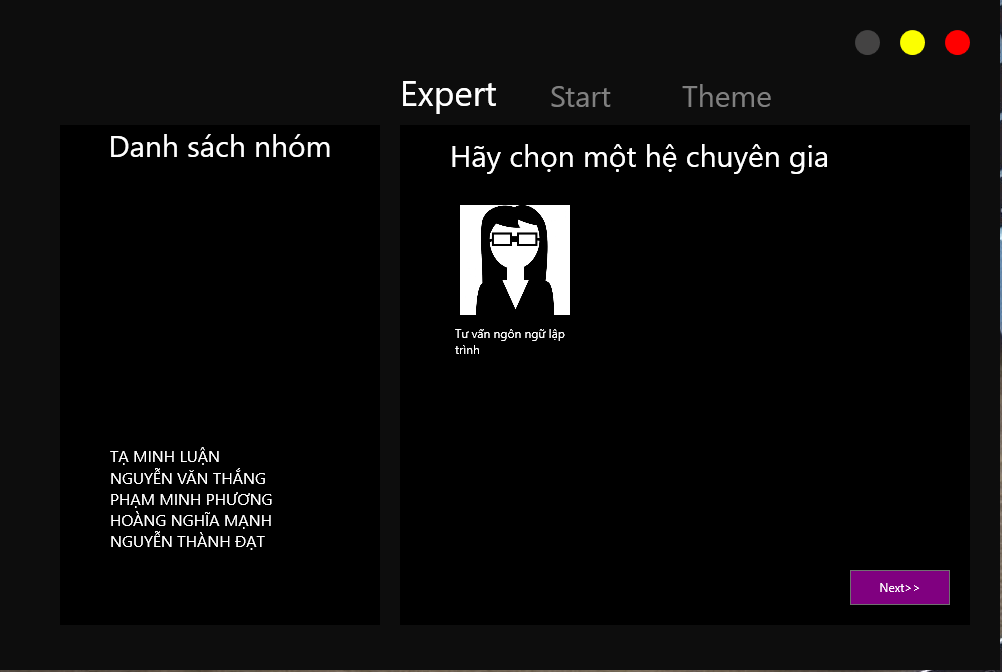
X là con của Z và answer(Z,Y) đúng

answer(X,Y):-answer(Z,Y),subset(X,Z).

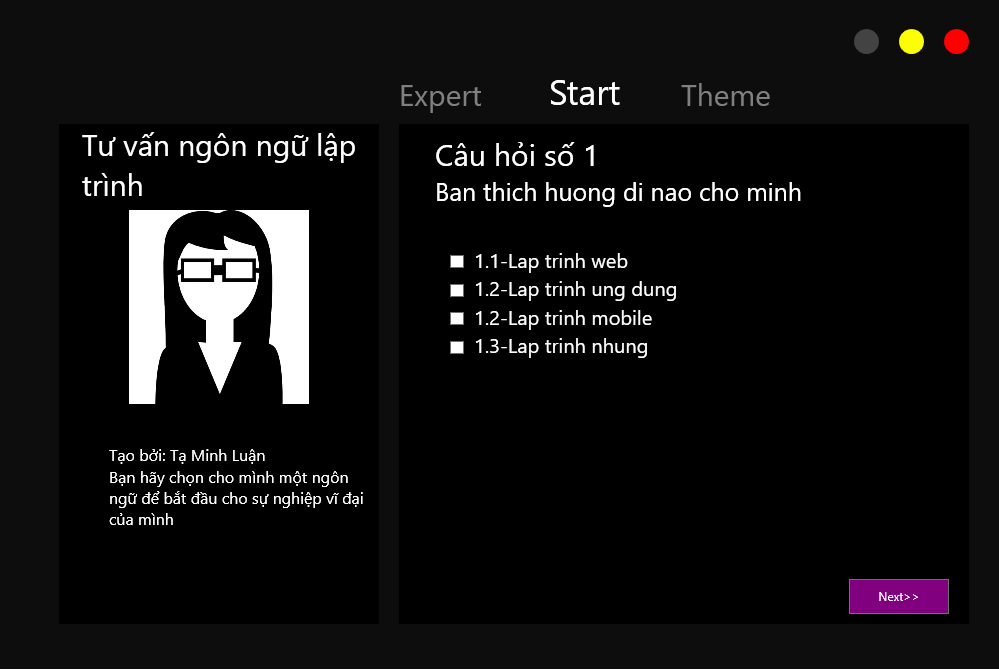
### 3.2.3 File prolog



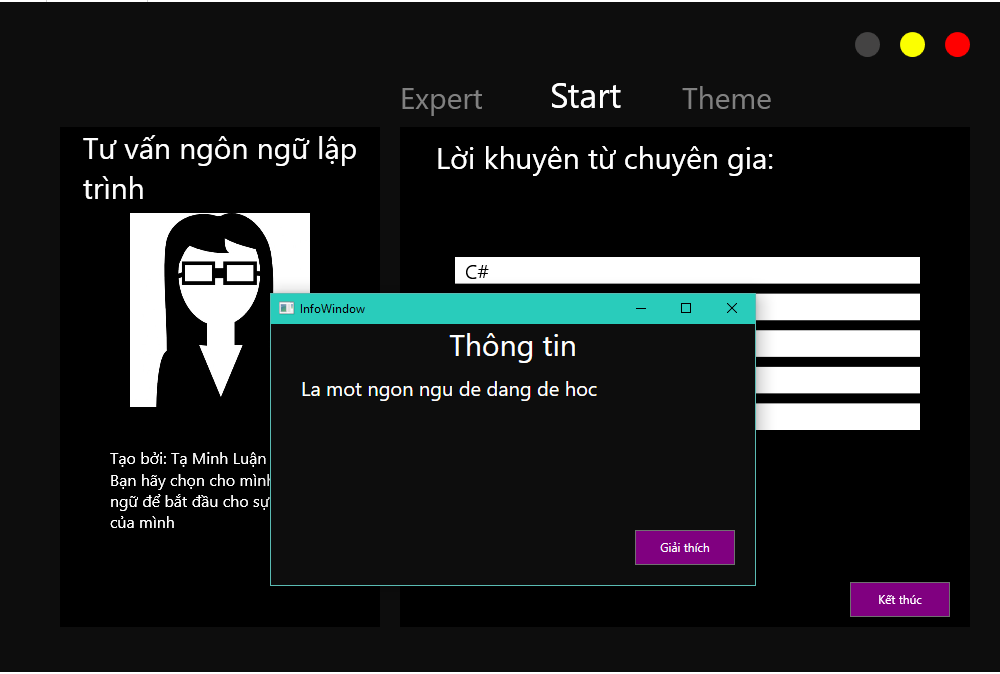
## 3.3 Hệ chuyên gia tư vấn ngôn ngữ lập trình



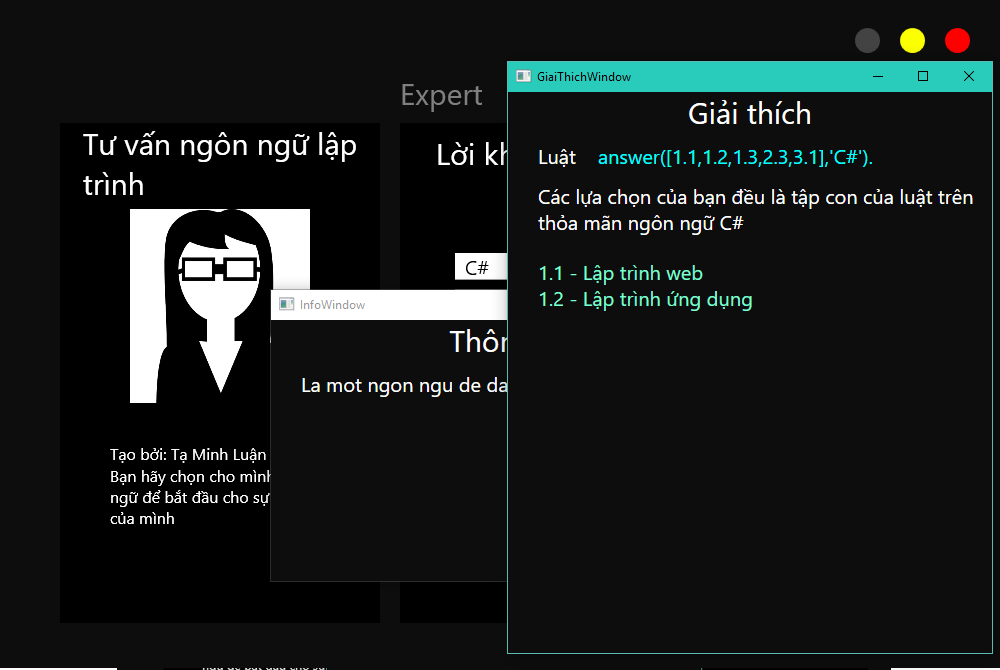
*Hình 3.1: Giao diện khởi động của hệ chuyên gia*



*Hình 3.2: Các câu hỏi từ hệ chuyên gia*



*Hình 3.3: Lời khuyên từ chuyên gia và thông tin của các ngôn ngữ*



*Hình 3.3: Chuyên gia đưa ra lời giải thích*