

TRƯỜNG ĐẠI HỌC VINH

VIỆN KỸ THUẬT VÀ CÔNG NGHỆ



ĐỒ ÁN KỸ THUẬT LẬP TRÌNH NGÀNH KỸ THUẬT ĐIỀU KHIỂN VÀ TỰ ĐỘNG HÓA

ĐỀ TÀI: THIẾT KẾ TRỢ LÝ ẢO AI BẰNG NGÔN NGỮ LẬP TRÌNH PYTHON

Giảng viên hướng dẫn : TS. Mai Thế Anh

Sinh viên thực hiện : Nguyễn Thế Cường

Lớp : 55K1 KTĐK & TĐH

NGHỆ AN, 2021

MỤC LỤC

Contents

CHƯƠNG 1 : CƠ SỞ LÝ THUYẾT	5
1. Tổng quan về AI.....	5
<i>1.1 Khái niệm cơ bản về AI :</i>	<i>5</i>
<i>1.2. Sự cần thiết việc học AI :.....</i>	<i>5</i>
<i>1.3. Thông minh là gì?</i>	<i>6</i>
<i>1.4. Thông minh được cấu thành từ gì?</i>	<i>7</i>
<i>1.5. Các vấn đề liên quan đến AI :</i>	<i>8</i>
2. Ngôn ngữ lập trình.....	10
3. Ứng dụng Sublime Text 3.....	11
CHƯƠNG 2 : LẬP TRÌNH CHƯƠNG TRÌNH TRÍ TUỆ NHÂN TẠO AI... 14	14
1. Tổng quan về ngôn ngữ lập trình Python.....	14
1.1 Python là gì ?	14
1.2 Cài đặt Python.....	15
2. Thiết kế chương trình.....	17
2.1 Sơ đồ tư duy	18
2.2 Viết chương trình trợ lý ảo Ai.....	20
3. Kết quả thực tế	22
KẾT LUẬN	23

MỤC LỤC HÌNH ẢNH

<i>Hình 1.1 Giao diện Sublime Text 3</i>	<i>13</i>
<i>Hình 2.1 Sơ đồ khối của chương trình</i>	<i>18</i>
<i>Hình 2.2 Lưu đồ thuật toán của chương trình</i>	<i>19</i>
<i>Hình 2.3 Chương trình lập trình phân hiệu</i>	<i>20</i>
<i>Hình 2.4 Chương trình phân nghe</i>	<i>20</i>
<i>Hình 2.5 Chương trình phân nói</i>	<i>21</i>
<i>Hình 2.6 Kết quả thực hiện chương trình.....</i>	<i>22</i>

LỜI NÓI ĐẦU

Gần đây, Trợ lý ảo đã có được nền tảng chính sau khi Apple tích hợp Trợ lý ảo đáng kinh ngạc nhất - Siri, chính thức là một phần của Apple Inc. Nhưng dòng thời gian của sự tiến hóa vĩ đại nhất bắt đầu từ sự kiện năm 1962 tại Hội chợ Thế giới Seattle nơi IBM trưng bày bộ máy đọc đảo gọi là Shoebox. Đó là kích thước thực tế của một hộp đựng giày và có thể thực hiện các chức năng khoa học và có thể nhận biết 16 từ và cũng có thể nói chúng bằng giọng nói dễ nhận biết của con người với 0 đến 9 chữ số.

Vào tháng 4 năm 1997, Dragon NataturalSpeaking là phần mềm chỉnh sửa chính tả đầu tiên có thể hiểu khoảng 100 từ và biến nó thành nội dung có thể đọc được. Theo dòng thời gian, xử lý ngôn ngữ tự nhiên và trợ lý ảo được đầu tư nghiên cứu mạnh mẽ, trải qua những cột mốc quan trọng và đạt được thành tựu thông minh như ngày hôm nay. Sau một thời gian học tập và nghiên cứu ở trường, em đã được làm quen với các môn học lập trình. Để áp dụng lý thuyết với thực tế học kỳ này em xin làm đề tài về “Thiết kế trợ lý ảo AI bằng ngôn ngữ lập trình Python”

Tuy nhiên do kiến thức chuyên môn còn hạn chế, tài liệu tham khảo có giới hạn nên còn xảy ra nhiều sai sót. Em rất mong thầy và các bạn góp ý bổ sung để bản đồ án của em được hoàn thiện hơn và giúp em hiểu biết hơn trong quá trình học tập tiếp theo.

Trong quá trình làm đồ án em đã nhận được sự giúp đỡ của các thầy giáo trong bộ môn và đặc biệt là sự chỉ bảo tận tình của thầy Mai Thế Anh đã giúp em hoàn thành đồ án này. Em xin chân thành cảm ơn thầy và mong thầy sẽ giúp đỡ chúng nữa trong việc học tập của em sau này..

Sinh viên thực hiện :

Nguyễn Thế Cường

CHƯƠNG 1 : CƠ SỞ LÝ THUYẾT

1. Tổng quan về AI

Kể từ khi phát minh ra máy tính hoặc máy móc, khả năng thực hiện các nhiệm vụ khác nhau của chúng đã tăng lên theo cấp số nhân. Con người đã phát triển sức mạnh của hệ thống máy tính về các lĩnh vực làm việc đa dạng của chúng, tốc độ ngày càng tăng và giảm kích thước theo thời gian.

Một nhánh của Khoa học Máy tính có tên Trí tuệ nhân tạo theo đuổi việc tạo ra các máy tính hoặc máy móc thông minh như con người.

1.1 Khái niệm cơ bản về AI :

Theo cha đẻ của Trí tuệ nhân tạo, John McCarthy, đó là “Khoa học và kỹ thuật tạo ra các máy thông minh, đặc biệt là các chương trình máy tính thông minh”.

Trí tuệ nhân tạo là một cách làm cho một máy tính, một robot điều khiển bằng máy tính hoặc một phần mềm suy nghĩ một cách thông minh, theo cách tương tự như suy nghĩ của con người. AI được hoàn thiện bằng cách nghiên cứu cách suy nghĩ của bộ não con người và cách con người học hỏi, quyết định và làm việc trong khi cố gắng giải quyết một vấn đề, sau đó sử dụng kết quả của nghiên cứu này làm cơ sở phát triển phần mềm và hệ thống thông minh.

Trong khi khai thác sức mạnh của hệ thống máy tính, sự tò mò của con người khiến anh ta tự hỏi, "Liệu một cỗ máy có thể suy nghĩ và hành xử như con người không?"

Do đó, sự phát triển của AI bắt đầu với mục đích tạo ra trí thông minh tương tự trong các máy móc mà chúng ta tìm thấy và đánh giá cao ở con người.

1.2. Sự cần thiết việc học AI :

Như chúng ta biết rằng AI theo đuổi việc tạo ra những cỗ máy thông minh như con người. Có rất nhiều lý do để chúng ta nghiên cứu về AI, ta cùng tìm xem qua một vài lý do :

AI có thể học thông qua dữ liệu :

Trong cuộc sống hàng ngày, chúng ta phải tiếp xúc với lượng dữ liệu khổng lồ và bộ não của con người không thể theo dõi được quá nhiều dữ liệu. Đó là lý do tại sao chúng ta cần tự động hóa mọi thứ. Để thực hiện tự động hóa, chúng ta cần nghiên cứu AI vì nó có thể học từ dữ liệu và có thể thực hiện các tác vụ lặp đi lặp lại một cách chính xác và không mệt mỏi.

AI có thể tự học :

Điều rất cần thiết là một hệ thống phải tự dạy vì bản thân dữ liệu luôn thay đổi và kiến thức có được từ dữ liệu đó phải được cập nhật liên tục. Chúng ta có thể sử dụng AI để thực hiện mục đích này vì hệ thống hỗ trợ AI có thể tự dạy.

AI có thể phản hồi trong thời gian thực:

Trí tuệ nhân tạo với sự trợ giúp của mạng nơ-ron có thể phân tích dữ liệu sâu hơn. Do khả năng này, AI có thể suy nghĩ và phản ứng với các tình huống dựa trên các điều kiện trong thời gian thực.

AI đạt được độ chính xác :

Với sự trợ giúp của mạng nơ-ron sâu, AI có thể đạt được độ chính xác cao. AI giúp trong lĩnh vực y học chẩn đoán các bệnh như ung thư từ MRI của bệnh nhân.

AI có thể tổ chức dữ liệu để tận dụng tối đa :

Dữ liệu là tài sản trí tuệ cho các hệ thống đang sử dụng các thuật toán tự học. Chúng ta cần AI để lập chỉ mục và tổ chức dữ liệu theo cách mà nó luôn mang lại kết quả tốt nhất.

Hiểu biết về trí thông minh :

Với AI, các hệ thống thông minh có thể được xây dựng. Chúng ta cần hiểu khái niệm trí thông minh để bộ não của chúng ta có thể xây dựng một hệ thống thông minh khác giống như chính nó.

1.3. Thông minh là gì?

Khả năng của một hệ thống để tính toán, suy luận, nhận thức các mối quan hệ và loại suy, học hỏi kinh nghiệm, lưu trữ và truy xuất thông tin từ bộ nhớ, giải

quyết vấn đề, hiểu các ý tưởng phức tạp, sử dụng ngôn ngữ tự nhiên thành thạo, phân loại, khái quát hóa và thích ứng với các tình huống mới.

Các loại trí thông minh

Theo mô tả của Howard Gardner, một nhà tâm lý học về phát triển người Mỹ, Trí thông minh có nhiều mặt -

Trí tuệ ngôn ngữ: Khả năng nói, nhận biết và sử dụng các cơ chế âm vị học (âm thanh lời nói), cú pháp (ngữ pháp) và ngữ nghĩa (ý nghĩa).

Trí tuệ âm nhạc : Khả năng tạo ra, giao tiếp và hiểu các ý nghĩa của âm thanh, hiểu cao độ, nhịp điệu.

Trí thông minh logic-toán học : Khả năng sử dụng và hiểu các mối quan hệ trong trường hợp không có hành động hoặc đối tượng. Đó cũng là khả năng hiểu những ý tưởng phức tạp và trừu tượng.

Trí tuệ không gian : Khả năng nhận thức thông tin hình ảnh hoặc thông tin không gian, thay đổi nó và tạo lại hình ảnh trực quan mà không cần tham chiếu đến các đối tượng, tạo hình ảnh 3D cũng như di chuyển và xoay chúng.

Trí thông minh thể chất : Khả năng sử dụng toàn bộ hoặc một phần cơ thể để giải quyết các vấn đề hoặc các sản phẩm thời trang, kiểm soát các kỹ năng vận động tinh và thô cũng như thao tác các đồ vật.

Trí tuệ intrapersonal : Khả năng phân biệt giữa cảm xúc, ý định và động cơ của chính một người.

Trí thông minh giữa các cá nhân : Khả năng nhận biết và phân biệt cảm xúc, niềm tin và ý định của người khác.

Bạn có thể nói một cỗ máy hoặc một hệ thống thông minh nhân tạo khi nó được trang bị ít nhất một hoặc tất cả các trí thông minh trong đó.

1.4. Thông minh được cấu thành từ gì?

Trí thông minh là vô hình. Nó bao gồm :

Lý luận

Học tập

Giải quyết vấn đề

Nhận thức

Trí tuệ ngôn ngữ

1.5. Các vấn đề liên quan đến AI :

Trí tuệ nhân tạo là một lĩnh vực nghiên cứu rộng lớn. Lĩnh vực nghiên cứu này giúp tìm ra giải pháp cho các vấn đề trong thế giới thực.

Bây giờ chúng ta hãy xem các lĩnh vực nghiên cứu khác nhau trong AI :

Học máy

Đây là một trong những lĩnh vực phổ biến nhất của AI. Khái niệm cơ bản của tập này là làm cho máy học từ dữ liệu khi con người có thể học hỏi từ kinh nghiệm của mình. Nó chứa các mô hình học tập trên cơ sở đó có thể đưa ra các dự đoán trên dữ liệu chưa biết.

Logic

Đây là một lĩnh vực nghiên cứu quan trọng khác, trong đó logic toán học được sử dụng để thực thi các chương trình máy tính. Nó chứa các quy tắc và dữ kiện để thực hiện đối sánh mẫu, phân tích ngữ nghĩa, v.v.

Searching

Lĩnh vực nghiên cứu này về cơ bản được sử dụng trong các trò chơi như cờ vua, tic-tac-toe. Các thuật toán tìm kiếm đưa ra giải pháp tối ưu sau khi tìm kiếm trên toàn bộ không gian tìm kiếm.

Artificial neural networks

Đây là một mạng lưới các hệ thống tính toán hiệu quả mà chủ đề trung tâm của nó được vay mượn từ sự tương tự của các mạng thần kinh sinh học. ANN có thể được sử dụng trong robot, nhận dạng giọng nói, xử lý giọng nói, v.v.

Thuật toán di truyền

Các thuật toán di truyền giúp giải quyết các vấn đề với sự hỗ trợ của nhiều chương trình. Kết quả sẽ dựa trên việc chọn người phù hợp nhất.

Knowledge Representation

Đây là lĩnh vực nghiên cứu với sự trợ giúp của chúng ta có thể biểu diễn các sự kiện theo cách mà máy móc có thể hiểu được. Tri thức được trình bày hiệu quả hơn; hệ thống càng thông minh.

6. Ứng dụng của AI :

Trong phần này, chúng ta sẽ thấy ứng dụng mà AI mang lại :

Game

AI đóng vai trò quan trọng trong các trò chơi chiến lược như cờ vua, poker, tic-tac-toe, v.v., nơi máy có thể nghĩ ra một số lượng lớn các vị trí có thể dựa trên kiến thức kinh nghiệm.

Xử lý ngôn ngữ tự nhiên

Có thể tương tác với máy tính hiểu ngôn ngữ tự nhiên của con người.

Expert Systems

Có một số ứng dụng tích hợp máy móc, phần mềm và thông tin đặc biệt để truyền đạt lý luận và tư vấn. Họ cung cấp giải thích và lời khuyên cho người dùng.

Vision systems

Các hệ thống này hiểu, diễn giải và lĩnh hội đầu vào trực quan trên máy tính. Ví dụ,

- Một chiếc máy bay do thám chụp ảnh, được sử dụng để tìm ra thông tin không gian hoặc bản đồ của các khu vực.
- Các bác sĩ sử dụng hệ thống chuyên gia lâm sàng để chẩn đoán cho bệnh nhân.
- Cảnh sát sử dụng phần mềm máy tính có thể nhận dạng khuôn mặt của tội phạm với bức chân dung được lưu trữ do nghệ sĩ pháp y thực hiện.

Nhận dạng giọng nói

Một số hệ thống thông minh có khả năng nghe và hiểu ngôn ngữ dưới dạng câu và ý nghĩa của chúng trong khi con người nói chuyện với nó. Nó có thể

xử lý các trọng âm khác nhau, từ lỏng, tiếng ồn trong nền, thay đổi tiếng ồn của con người do lạnh, v.v.

Nhận dạng chu Viet

Phần mềm nhận dạng chữ viết tay đọc văn bản được viết trên giấy bằng bút hoặc trên màn hình bằng bút cảm ứng. Nó có thể nhận ra hình dạng của các chữ cái và chuyển nó thành văn bản có thể chỉnh sửa.

Robot thông minh

Robot có thể thực hiện các nhiệm vụ do con người giao. Chúng có các cảm biến để phát hiện dữ liệu vật lý từ thế giới thực như ánh sáng, nhiệt, nhiệt độ, chuyển động, âm thanh, va đập và áp suất. Chúng có bộ xử lý hiệu quả, nhiều cảm biến và bộ nhớ khổng lồ, để thể hiện trí thông minh. Ngoài ra, họ có khả năng học hỏi từ những sai lầm của mình và họ có thể thích nghi với môi trường mới.

2. Ngôn ngữ lập trình

Trong bài học này, chúng ta bắt đầu với những khái niệm đầu tiên về ngôn ngữ lập trình:

- Ngôn ngữ lập trình là gì?
- Nó được sử dụng với mục đích gì?
- Có các loại ngôn ngữ lập trình nào?
- Làm thế nào để chọn một ngôn ngữ lập trình phù hợp với bản thân?

Ngày nay, những máy tính hiện đại có tốc độ xử lý nhanh đáng kinh ngạc, nhưng máy tính vốn chỉ có thể hiểu được một cách giới hạn các câu lệnh, chúng ta cần phải nói với máy tính chính xác những gì chúng ta muốn nó làm.

Một chương trình máy tính (hay còn được gọi là phần mềm hoặc ứng dụng) là một tập hợp các câu lệnh, chúng nói cho máy tính biết phải làm những gì, và các thiết bị trên máy tính thực thi các câu lệnh được gọi là phần cứng.

Những tập hợp các câu lệnh được thiết kế để ra lệnh cho máy tính được gọi là Ngôn ngữ lập trình.

Ngôn ngữ lập trình được chia là 2 loại:

- **Ngôn ngữ bậc thấp (Mã máy):** Một CPU của máy tính không đủ khả năng để hiểu được ngôn ngữ lập trình C++. Chỉ một tập hợp rất hạn chế các chỉ thị mà máy tính hiểu nhiên hiểu được, chúng được gọi là mã máy (ngôn ngữ máy hay tập chỉ thị). Hay nói cách khác, ngôn ngữ lập trình bậc thấp cung cấp cho máy tính các chỉ thị rõ ràng, không có tính trừu tượng giúp máy tính có thể hiểu được ngay lập tức.

- Một chỉ thị được hợp thành bởi một số các con số ở dạng nhị phân, nó chỉ có thể là 0 hoặc 1. Những con số dạng nhị phân này thường được gọi là những bit (là đơn vị lưu trữ nhỏ nhất trong máy tính).

- Mỗi tập hợp các số nhị phân được dịch bởi CPU thành các chỉ thị bảo máy tính làm một công việc cụ thể. Ví dụ: so sánh 2 số, lưu một giá trị vào một vùng nhớ máy tính. Mỗi CPU có một tập các chỉ thị khác nhau. Lập trình viên là những người tạo ra mã máy, nó rất khó và tốn thời gian để làm một thứ gì đó.

- **Ngôn ngữ lập trình bậc cao:** Ngôn ngữ lập trình bậc cao được thiết kế để lập trình viên viết chương trình mà không cần bận tâm vào loại chương trình máy tính đang chạy trên nó, có thể thấy ngôn ngữ lập trình bậc cao khá gần gũi với ngôn ngữ tự nhiên. Vì thế ngôn ngữ lập trình bậc cao phải được dịch ngược về mã máy để CPU có thể hiểu được chúng.

3. Ứng dụng Sublime Text 3

Là một IDE Editor hỗ trợ đa ngôn ngữ lập trình và hoàn toàn miễn phí. Nó có thể chạy được trên nhiều hệ điều hành khác nhau với cấu hình nhỏ gọn và hoàn toàn miễn phí. Sublime Text 3 là phiên bản mới nhất, trước đó có các phiên

bản 1 và 2. Với hệ thống plugin phong phú hay còn được gọi là control package giúp người dùng dễ dàng cài đặt thêm các tính năng hỗ trợ lập trình với ngôn ngữ mà mình chọn. Những ngôn ngữ mà công cụ này hỗ trợ mạnh nhất gồm HTML, CSS, JavaScript, C#, C++, Python...

Những tính năng chính của Sublime Text

Người dùng có thể tùy chỉnh nhiều tính năng

Các ràng buộc chính, menu, đoạn trích, macro và nhiều thứ khác trong Sublime Text đều có thể tùy chỉnh với các tệp JSON đơn giản. Hệ thống này mang đến cho bạn sự linh hoạt vì các cài đặt có thể được chỉ định trên loại file và từng dự án khác nhau.

Hiệu suất cao

Sublime Text được xây dựng từ các thành phần tùy chỉnh, cung cấp khả năng đáp ứng chưa từng có. Từ bộ công cụ UI đa nền tảng tùy chỉnh mạnh mẽ, đến công cụ tô sáng cú pháp. Đây được xem là IDE có hiệu suất nhanh và ổn định nhất hiện nay.

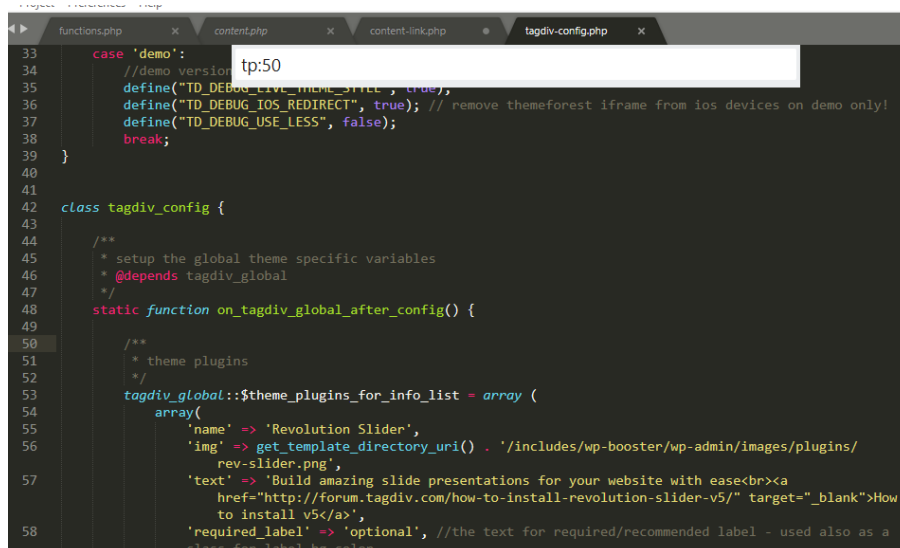
Hiện thị tất cả các phím tắt Sublime Text

Phần mềm này mạnh mẽ vì sử dụng rất nhiều phím tắt giúp thao tác viết code nhanh hơn. Để xem tất cả các phím tắt của phần mềm hay các package đã cài đặt bạn nhấn phím tắt Ctrl+Shift + P. Để kết thúc bản hệ thống phím tắt bạn nhấn tổ hợp phím này lại nha.

Tìm nhanh hàm và các câu lệnh trong 1 file

Nhiều dự án bạn viết rất nhiều hàm hoặc câu lệnh trong 1 file. Để tìm chúng bạn hãy sử dụng phím tắt Ctrl +P.

Nhập @ để chuyển đến hàm đầu tiên của file, phím # để tìm kiếm câu lệnh hoặc hàm bất kỳ nào đó. Nếu muốn tìm kiếm vị trí dòng cụ thể bạn nhấn phím tắt tp:số dòng, ví dụ tìm dòng 50 nhấn tp:50.



Hình 1.1 Giao diện Sublime Text 3

Hỗ trợ API mạnh mẽ

Sublime Text có API Python mạnh mẽ, cho phép các plugin tăng cường chức năng tích hợp. Kiểm soát gói có thể được cài đặt thông qua bảng lệnh, cung cấp quyền truy cập đơn giản vào hàng ngàn gói được xây dựng bởi cộng đồng.

Chia màn hình viết code thành nhiều cột

Để dễ dàng so sánh code của nhiều file cùng lúc, bạn có thể chia màn hình chính thành nhiều cột khác nhau. Vào View -> Layout. Tính năng này giúp bạn có cái nhìn trực quan hơn về các đoạn code mà mình muốn tìm hiểu.

cách sử dụng Sublime text 3

Hiển thị và tìm kiếm các hàm giống nhau

Để kiểm tra xem một hàm hay một lệnh được sử dụng bao nhiêu lần trên Sublime Text, bạn sử dụng chuột để bôi đen đoạn mã đó. Tiếp đến sử dụng phím tắt Ctrl +D để di chuyển và xem từng đoạn code nằm ở vị trí nào.

Hỗ trợ đa nền tảng











Sublime Text có sẵn cho Mac, Windows và Linux. Một giấy phép là tất cả những gì bạn cần để sử dụng Sublime Text trên mọi máy tính bạn sở hữu, bất kể nó sử dụng hệ điều hành nào. Sublime Text sử dụng bộ công cụ UI tùy chỉnh, được tối ưu hóa cho tốc độ, đồng thời tận dụng chức năng gốc trên mỗi nền tảng.

CHƯƠNG 2 : LẬP TRÌNH CHƯƠNG TRÌNH TRÍ TUỆ NHÂN TẠO AI

1. Tổng quan về ngôn ngữ lập trình Python

1.1 Python là gì ?

Python là một ngôn ngữ lập trình kịch bản (scripting language) do **Guido van Rossum** tạo ra năm 1990. Đến nay thì cộng đồng người sử dụng ngôn ngữ này rất đông, nếu so sánh từ bảng xếp hạng các ngôn ngữ năm 2016 thì Python đứng thứ 3 trong top 10 ngôn ngữ phổ biến nhất.

Language Rank	Types	Spectrum Ranking
1. C		100.0
2. Java		98.1
3. Python		98.0
4. C++		95.9
5. R		87.9
6. C#		86.7
7. PHP		82.8
8. JavaScript		82.2
9. Ruby		74.5
10. Go		71.9

Và Python luôn được biết đến với các đặc điểm sau đây:

- Cú pháp rất tường minh, dễ đọc.
- Các khả năng tự xét mạnh mẽ.
- Hướng đối tượng trực giác.
- Cách thể hiện tự nhiên mã thủ tục.
- Hoàn toàn mô-đun hóa, hỗ trợ các gói theo cấp bậc.
- Xử lý lỗi dựa theo ngoại lệ.
- Kiểu dữ liệu động ở mức rất cao.
- Các thư viện chuẩn và các mô-đun ngoài bao quát hầu như mọi việc.
- Phần mở rộng và mô-đun dễ dàng viết trong C, C++.

- Có thể nhúng trong ứng dụng như một giao diện kịch bản (scripting interface).
- Python mạnh mẽ và thực hiện rất nhanh.

1.2 Cài đặt Python.

Python hỗ trợ hầu hết trên các hệ điều hành hiện nay, và cách cài đặt nó cũng rất đơn giản. Tuy nhiên ở bài này mình sẽ chỉ hướng dẫn mọi người cài đặt python trên môi trường window thôi.

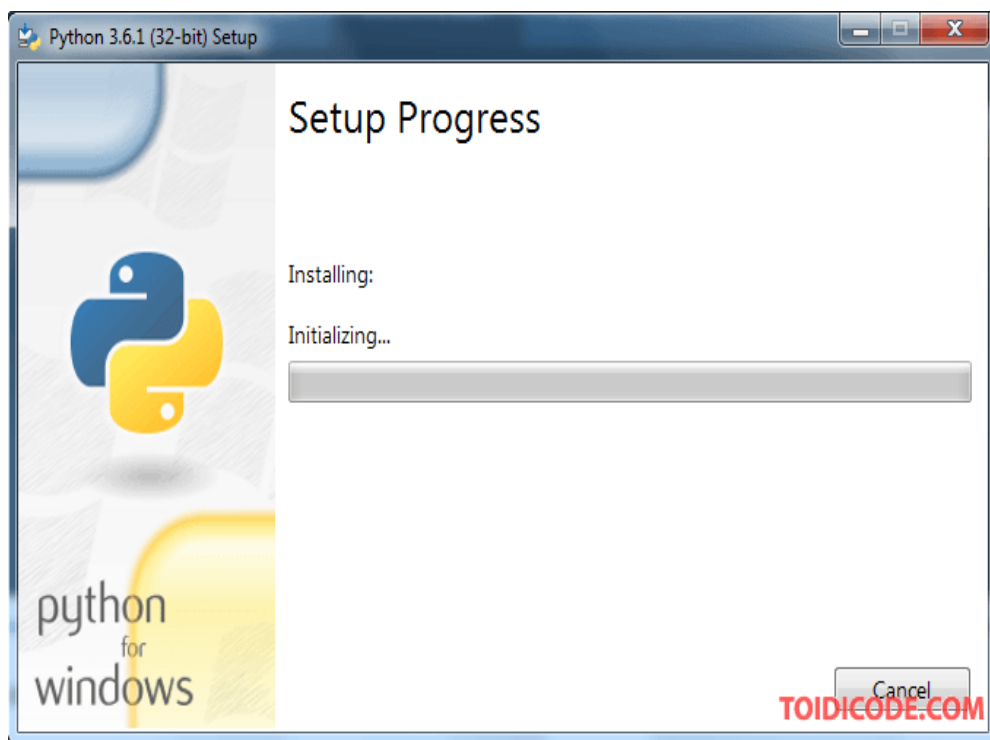
Đầu tiên chúng ta cần download bộ cài python từ trang chủ của nó ([download](#)). Ở đây mình chọn phiên bản mới nhất là phiên bản 3.6.1.



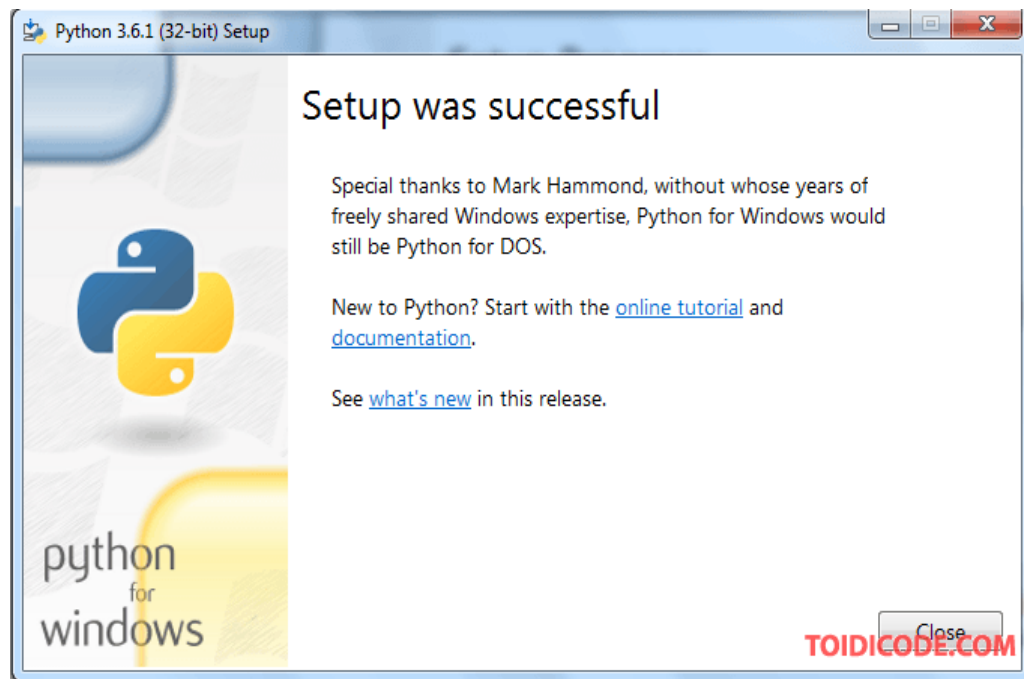
Tiếp đó chúng ta chạy file vừa tải về và cài đặt theo các bước như sau:



Tích chọn add python x.x to PATH->install Now

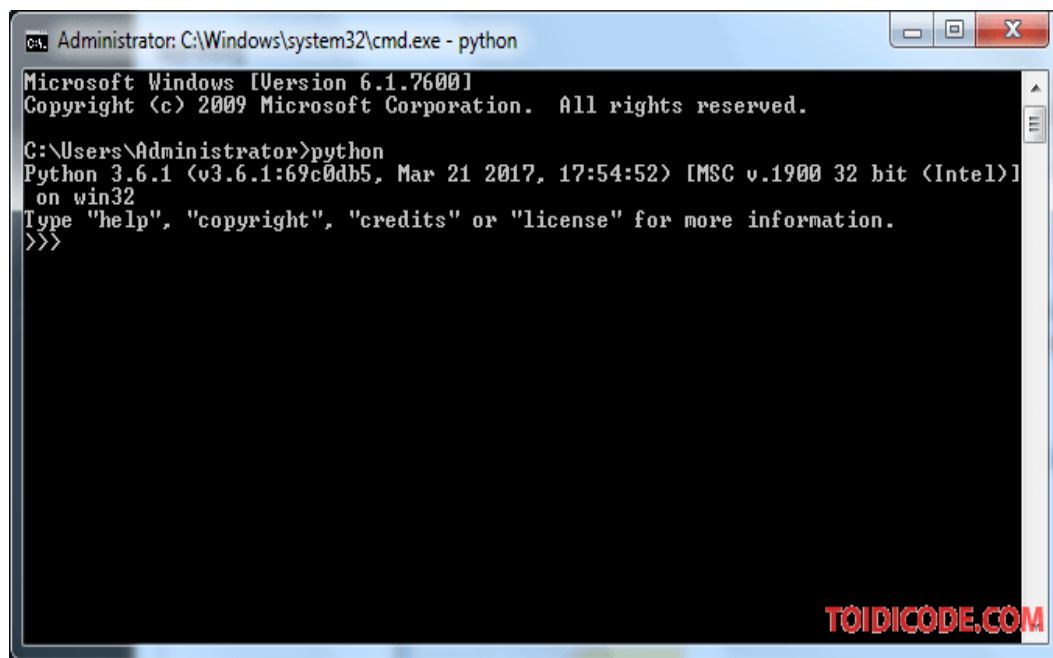


Đợi quá trình cài đặt



Màn hình báo thành công.

-Để chắc chắn lại là đã thành công hay chưa các bạn mở cmd lên và gõ lệnh **python**.



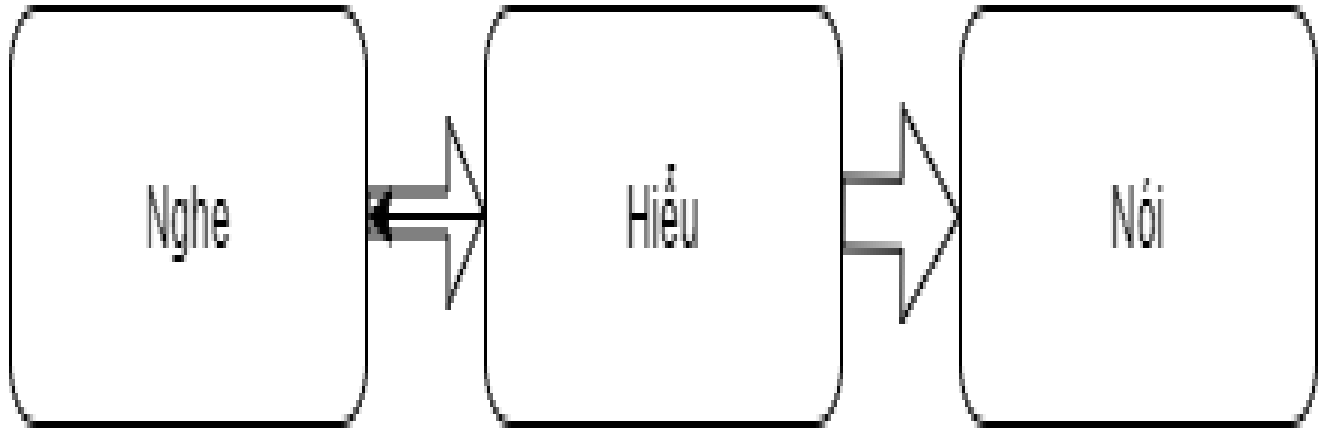
Nếu như xuất hiện cửa sổ như hình thì bạn đã cài đặt thành công Python

2. Thiết kế chương trình

Mục đích của đề tài : Viết một chương trình sử dụng ngôn ngữ Python thực hiện nhiệm vụ là trở thành một trợ lý Ai với chức năng trả lời những câu hỏi mà người hỏi đưa ra thông qua mic và loa.

2.1 Sơ đồ tư duy

Sơ đồ khối



Hình 2.1 Sơ đồ khối của chương trình

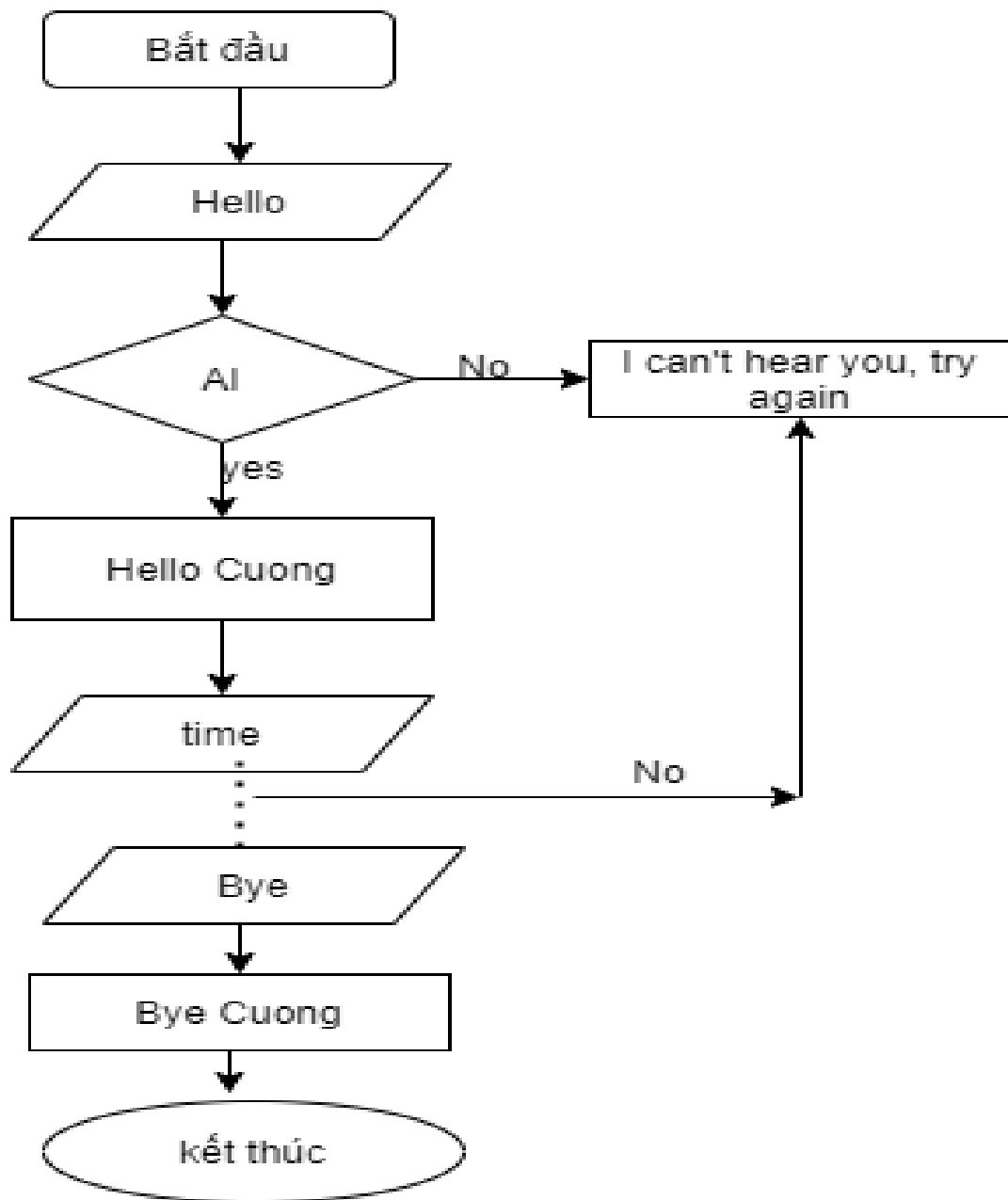
Giải thích :

Phần nghe : Tức máy tính biết được bạn đang nói gì thông qua mic của laptop sau đó nó thực hiện thao tác chuyển hóa thứ nó nghe được thành dạng chữ .

Phần hiểu : Sau khi có dạng chữ máy tính tìm kiếm câu trả lời ở dạng chữ khác (phần này mình setting sẵn câu trả lời).

Phần nói : Máy tính đọc câu trả lời thông qua loa ngoài của laptop.

Lưu đồ thuật toán

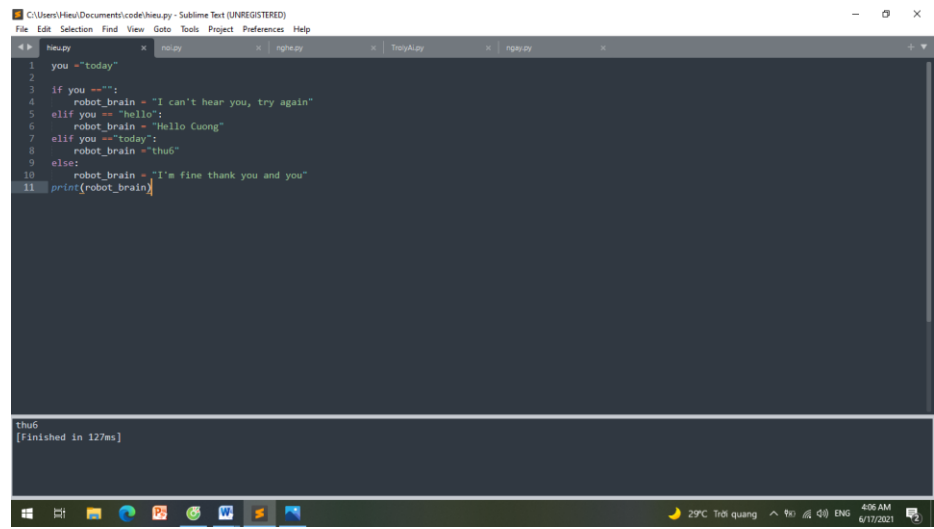


Hình 2.2 Lưu đồ thuật toán của chương trình

Giải thích : Bắt đầu, ta chạy chương trình bằng cmd .Sau khi mình hỏi Hello, máy tính sẽ nhận biết nếu nó nghe thấy từ Hello thì sẽ trả lời lại cho mình là Hello Cuong.Nếu không nghe được câu của mình hoặc câu nói khác máy tính sẽ hiểu là không có câu hỏi nào đặt ra cho nó : I can't hear you, try again. Yêu cầu mình hỏi lại nó. Tương tự với các câu hỏi tiếp theo. Để kết thúc chương trình mình nói Bye.

2.2 Viết chương trình trợ lý ảo Ai

Lập trình phản hồi :



```
1 you = "today"
2
3 if you == "":
4     robot_brain = "I can't hear you, try again"
5 elif you == "hello":
6     robot_brain = "Hello Cuong"
7 elif you == "today":
8     robot_brain = "thu6"
9 else:
10    robot_brain = "I'm fine thank you and you"
11 print(robot_brain)
```

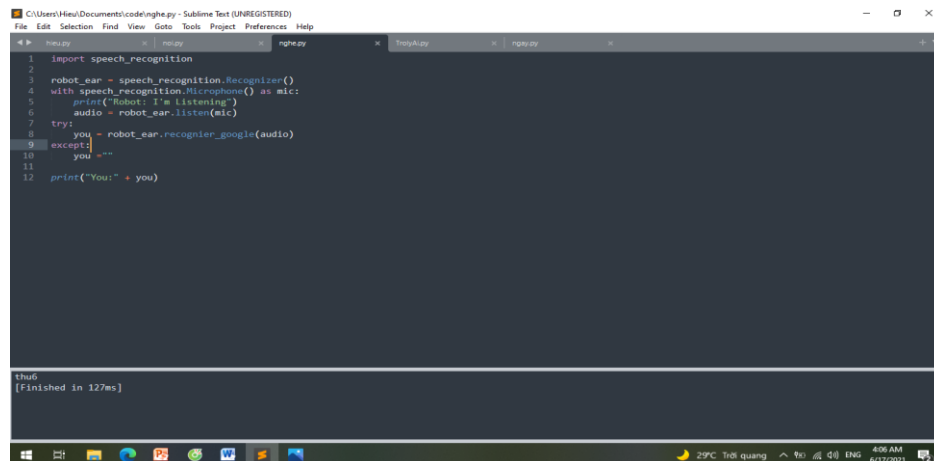
thu6
[Finished in 127ms]

Hình 2.3 Chương trình lập trình phản hồi

Ở phần này mình sẽ thực hiện mục đích là thực hiện việc giao tiếp với máy tính bằng dạng chữ. Tức mình đưa ra câu hỏi, máy tính trả lời.

Phần này mình gọi biến You là bản thân, đưa ra câu hỏi còn Robot_brain là trợ lý Ai đưa ra câu trả lời. Ví dụ mình hỏi Hello máy tính sẽ trả lời lại là Hello Cuong.

Lập trình phần nghe :



```
1 import speech_recognition
2
3 robot_ear = speech_recognition.Recognizer()
4 with speech_recognition.Microphone() as mic:
5     print("Robot: I'm listening")
6     audio = robot_ear.listen(mic)
7 try:
8     you = robot_ear.recognize_google(audio)
9 except:
10    you = ""
11 print("You:" + you)
```

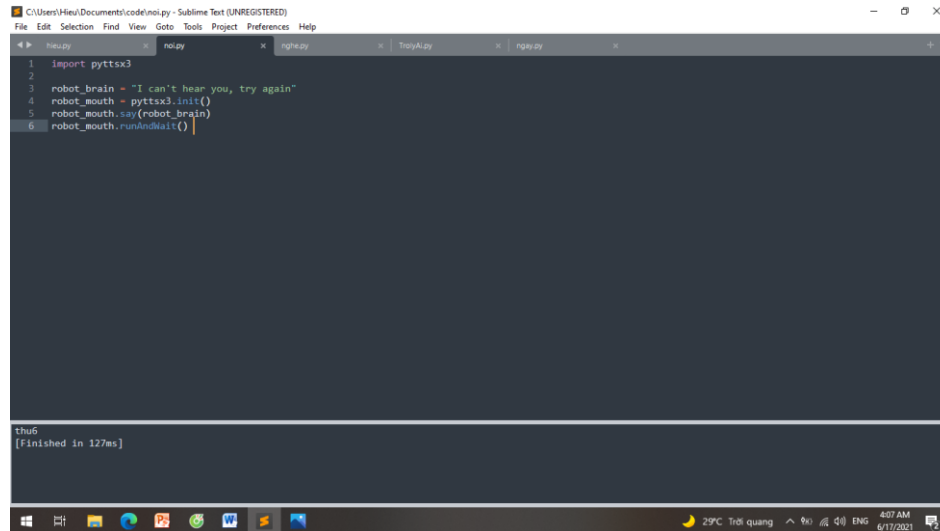
thu6
[Finished in 127ms]

Hình 2.4 Chương trình phần nghe

Ở phần này mình sẽ lập trình phần nghe, thực hiện nếu mình nói thì máy tính sẽ trả về dạng chữ cho mình. Đặt biến nghe là robot_ear.

Khi chạy lỗi mình dung hàm Try ... except để bắt lỗi nếu mình nói máy tính không nghe được và nó sẽ hiểu là you rỗng.

Lập trình phần nói :



```
1 import pyttsx3
2
3 robot_brain = "I can't hear you, try again"
4 robot_mouth = pyttsx3.init()
5 robot_mouth.say(robot_brain)
6 robot_mouth.runAndWait()
```

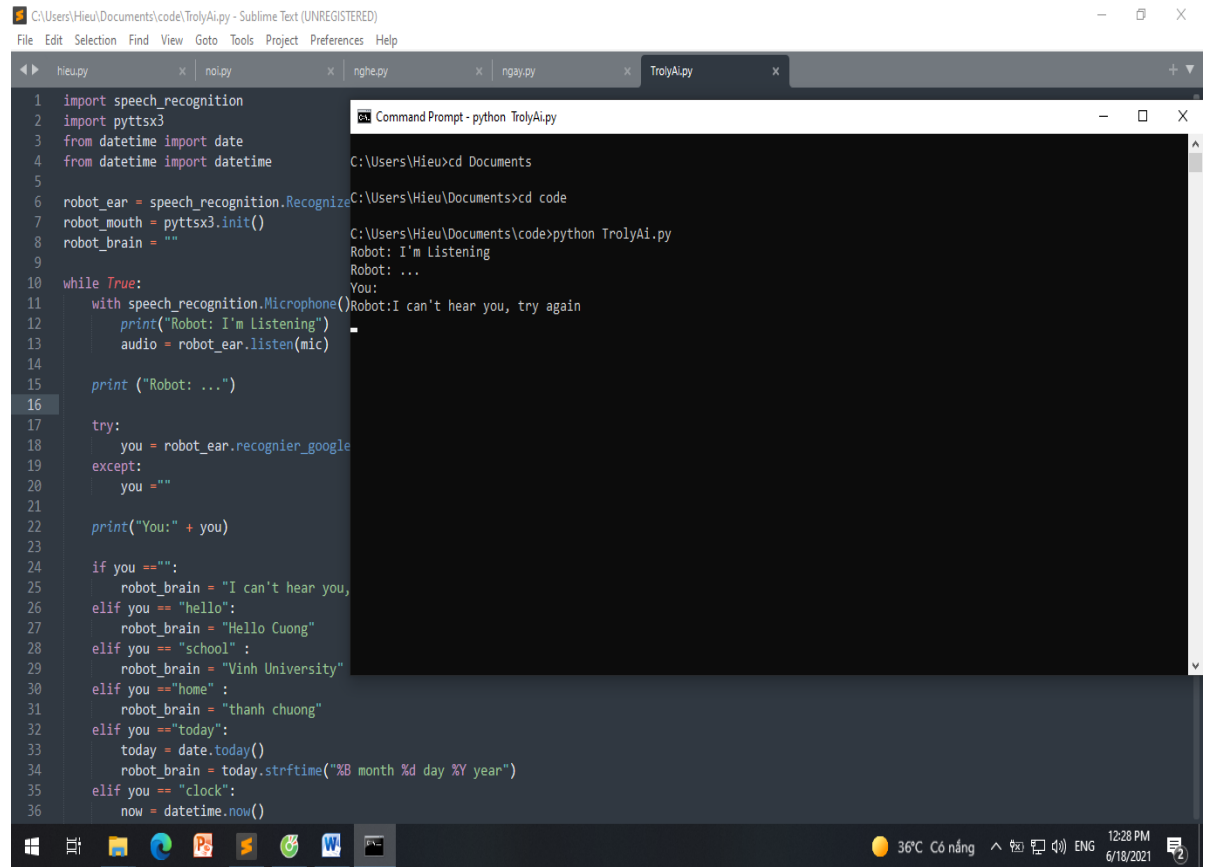
Thu6
[Finished in 127ms]

Hình 2.5 Chương trình phần nói

Phần này mình chuyển đổi giọng nói qua dạng văn bản hay dạng chữ. Ở phần này mình phải cài đặt thư viện nghe bằng cách gõ pip install speechrecognition. Mình đặt tên cho biến nói là robot_mouth.

Chương trình trợ lý ảo AI là gộp 3 phần trên lại với nhau.

3. Kết quả thực tế



```
1 import speech_recognition
2 import pyttsx3
3 from datetime import date
4 from datetime import datetime
5
6 robot_ear = speech_recognition.Recognize
7 robot_mouth = pyttsx3.init()
8 robot_brain = ""
9
10 while True:
11     with speech_recognition.Microphone() as mic:
12         print("Robot: I'm Listening")
13         audio = robot_ear.listen(mic)
14
15     print ("Robot: ...")
16
17     try:
18         you = robot_ear.recognizer_google(audio)
19     except:
20         you = ""
21
22     print("You:" + you)
23
24     if you == "":
25         robot_brain = "I can't hear you, try again"
26     elif you == "hello":
27         robot_brain = "Hello Cuong"
28     elif you == "school":
29         robot_brain = "Vinh University"
30     elif you == "home":
31         robot_brain = "thanh chuong"
32     elif you == "today":
33         today = date.today()
34         robot_brain = today.strftime("%B month %d day %Y year")
35     elif you == "clock":
36         now = datetime.now()
```

```
C:\Users\Hieu>cd Documents
C:\Users\Hieu\Documents>cd code
C:\Users\Hieu\Documents\code>python TrolleyAi.py
Robot: I'm Listening
Robot: ...
You: ...
Robot:I can't hear you, try again
```

Hình 2.6 Kết quả thực hiện chương trình

KẾT LUẬN

Sau một thời gian tìm hiểu và nghiên cứu em đã hoàn thành tốt đề tài được giao, hoàn thiện sơ đồ khối và chương trình thực tế .

Với nhiều khó khăn ban đầu khi mới bắt tay vào hoàn thiện sản phẩm với kiến thức ít ỏi của mình và kinh nghiệm làm chương trình code thực tế không có nhiều nên việc hoàn thiện đề tài gặp nhiều khó khăn .

Với nỗ lực của bản thân và đặc biệt là với sự hướng dẫn của thầy Mai Thế Anh em đã hoàn thành sản phẩm đúng thời gian. Trong quá trình hoàn thiện sản phẩm mạch có nhưng ưu điểm và nhược điểm sau

a. Ưu điểm

- Chương trình chạy khá ổn và trơn tru. Thực hiện hết được các chức năng yêu cầu.
- Phần nói khá rõ ràng và mạch lạc.

b. Nhược điểm và mở rộng đề tài

- Thư viện hiểu biết của Ai chương trình trên còn hẹp việc này có thể bổ sung bằng các dữ liệu nhập vào.
- Mới chỉ tích hợp tiếng Anh nên em muốn làm thêm về phần tiếng việt.
- Phần mic thu kém thì máy sẽ khó nhận dạng giọng nói khá khó khăn.

TÀI LIỆU THAM KHẢO

1. Võ Duy Tuấn, Python rất là cơ bản – Python Tiếng Việt (2017) Link ebook : <https://cuongquach.com/tai-lieu-python-tieng-viet-co-ban-vo-duy-tuan.html>
2. Tổng quan về Ai . Link wed : <https://insights.magestore.com/posts/tong-quan-ai-artificial-intelligence>
3. Tổng quan AI - Lập trình AI bằng Python .Link wed : <https://vncoder.vn/bai-hoc/tong-quan-ai-397>

Code :

```
import speech_recognition

import pytsx3

from datetime import date

from datetime import datetime

robot_ear = speech_recognition.Recognizer()

robot_mouth = pytsx3.init()

robot_brain = ""

while True:

    with speech_recognition.Microphone() as mic:

        print("Robot: I'm Listening")

        audio = robot_ear.listen(mic)

    print ("Robot: ...")

    try:

        you = robot_ear.recognier_google(audio)

    except:

        you = ""

    print("You:" + you)
```

```
if you == "":
    robot_brain = "I can't hear you, try again"

elif you == "hello":
    robot_brain = "Hello Cuong"

elif you == "school" :
    robot_brain = "Vinh University"

elif you == "home" :
    robot_brain = "thanh chuong"

elif you == "today":
    today = date.today()

    robot_brain = today.strftime("%B month %d day %Y year")

elif you == "clock":
    now = datetime.now()

    robot_brain = now.strftime("%H hours %M minutes %S seconds")

elif you == "bye":
    robot_brain = "Bye Cuong"

    print("Robot:" + robot_brain)

    robot_mouth.say(robot_brain)

    robot_mouth.runAndWait()

    break
```

else:

robot_brain = "I'm fine thank you and you"

print("Robot:" + robot_brain)

robot_mouth.say(robot_brain)

robot_mouth.runAndWait()