BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC NHA TRANG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN**

**TRÍ TUỆ NHÂN TẠO**

**Giảng viên hướng dẫn:** Nguyễn Đình Cường

**Sinh viên thực hiện:** Trần Văn Huy

**Mã số sinh viên:** 61131815

**Lớp học phần:** HETTNT

Khánh Hòa – 8/2022

GIỚI THIỆU

Trí tuệ nhân tạo hay trí thông minh nhân tạo (Artificial intelligence – viết tắt là AI) là một ngành thuộc lĩnh vực khoa học máy tính (Computer science). Là trí tuệ do con người lập trình tạo nên với mục tiêu giúp máy tính có thể tự động hóa các hành vi thông minh như con người hoặc thậm chí là hơn con người.

Trí tuệ nhân tạo khác với việc lập trình logic trong các ngôn ngữ lập trình là ở việc ứng dụng các hệ thống học máy (machine learning) để mô phỏng trí tuệ của con người trong các xử lý mà con người làm tốt hơn máy tính.

Cụ thể, trí tuệ nhân tạo giúp máy tính có được những trí tuệ của con người như: biết suy nghĩ và lập luận để giải quyết vấn đề, biết giao tiếp do hiểu ngôn ngữ, tiếng nói, biết học và tự thích nghi,…

Tuy rằng trí thông minh nhân tạo có nghĩa rộng như là trí thông minh trong các tác phẩm khoa học viễn tưởng, nó là một trong những ngành trọng yếu của tin học. Trí thông minh nhân tạo liên quan đến cách cư xử, sự học hỏi và khả năng thích ứng thông minh của máy móc.

MỤC LỤC

[I. Cài đặt bài toán Big Number 6](#_Toc111122455)

[1. Cơ sở lý thuyết 6](#_Toc111122456)

[2. Cài đặt 6](#_Toc111122457)

[3. Kết quả 12](#_Toc111122458)

[II. Cài đặt thuật toán Suy dẫn tiến 12](#_Toc111122459)

[1. Cơ sở lý thuyết 12](#_Toc111122460)

[2. Cài đặt chương trình 12](#_Toc111122461)

[3. Kết quả 15](#_Toc111122462)

[III. Cài đặt game Bóc sỏi 16](#_Toc111122463)

[1. Cơ sở lý thuyết 16](#_Toc111122464)

[2. Cài đặt chương trình 16](#_Toc111122465)

[3. Kết quả 17](#_Toc111122466)

[IV. Cài đặt game Tic-tac-toe 17](#_Toc111122467)

[1. Cơ sở lý thuyết 17](#_Toc111122468)

[2. Cài đặt chương trình 18](#_Toc111122469)

[3. Kết quả 27](#_Toc111122470)

[V. IoT – đồng hồ thời gian thực 28](#_Toc111122471)

[1. Cở sở lý thuyết 28](#_Toc111122472)

[2. Cài đặt chương trình 29](#_Toc111122473)

[3. Kết quả 31](#_Toc111122474)

[VI. Cài đặt bài toán 8 quân hậu 33](#_Toc111122475)

[1. Cở sở lý thuyết 33](#_Toc111122476)

[2. Cài đặt chương trình 33](#_Toc111122477)

[3. Kết quả 39](#_Toc111122478)

[VII. Cài đặt thuật toán Robinson 39](#_Toc111122479)

[1. Cơ sở lý thuyết 39](#_Toc111122480)

[2. Cài đặt chương trình 40](#_Toc111122481)

[3. Kết quả 46](#_Toc111122482)

[VIII. Cài đặt thuật toán Vương hạo 46](#_Toc111122483)

[1. Cơ sở lý thuyết 46](#_Toc111122484)

[2. Cài đặt chương trình 47](#_Toc111122485)

[3. Kết quả 52](#_Toc111122486)

[IX. Cài đặt Bài toán tam giác 52](#_Toc111122487)

[1. Cơ sở lý thuyết 52](#_Toc111122488)

[2. Cài đặt chương trình 53](#_Toc111122489)

[3. Kết quả 61](#_Toc111122490)

[X. Cài đặt thuật toán di truyền 61](#_Toc111122491)

[1. Cơ sở lý thuyết 61](#_Toc111122492)

[2. Cài đặt chương trình 64](#_Toc111122493)

[XI. Cài đặt thuật toán Heuristic 65](#_Toc111122494)

[1. Cơ sở lý thuyết 65](#_Toc111122495)

[2. Cài đặt chương trình 66](#_Toc111122496)

[3. Kết quả 69](#_Toc111122497)

[LINK SOURCE CODE 70](#_Toc111122498)

[1. Bài toàn Big Number 70](#_Toc111122499)

[2. Thuật toán Suy dẫn tiến 70](#_Toc111122500)

[3. Game Bóc sỏi 70](#_Toc111122501)

[4. Game Tic-tac-toe 70](#_Toc111122502)

[5. IoT - Đồng hồ thời gian thực 70](#_Toc111122503)

[6. Bài toán N-queen 70](#_Toc111122504)

[7. Thuật toán Robinson 70](#_Toc111122505)

[8. Thuật toán Vương hạo 70](#_Toc111122506)

[9. Bài toán tam giác 70](#_Toc111122507)

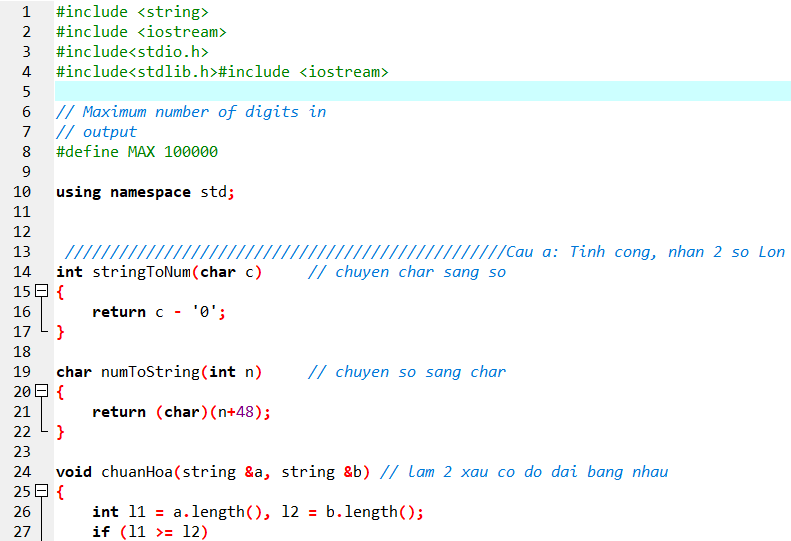
[10. Thuật toán di truyền 71](#_Toc111122508)

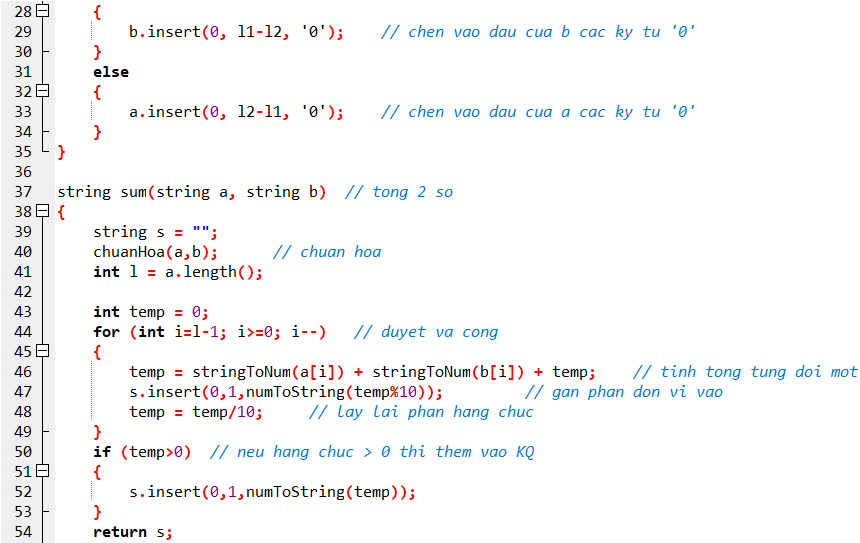
[11. Thuật toán Heuristic 71](#_Toc111122509)

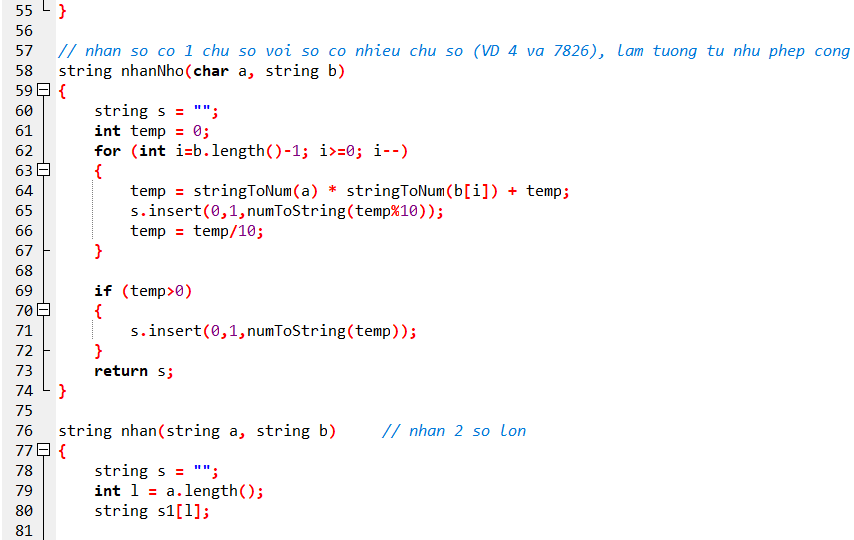
1. Cài đặt bài toán Big Number
2. Cơ sở lý thuyết

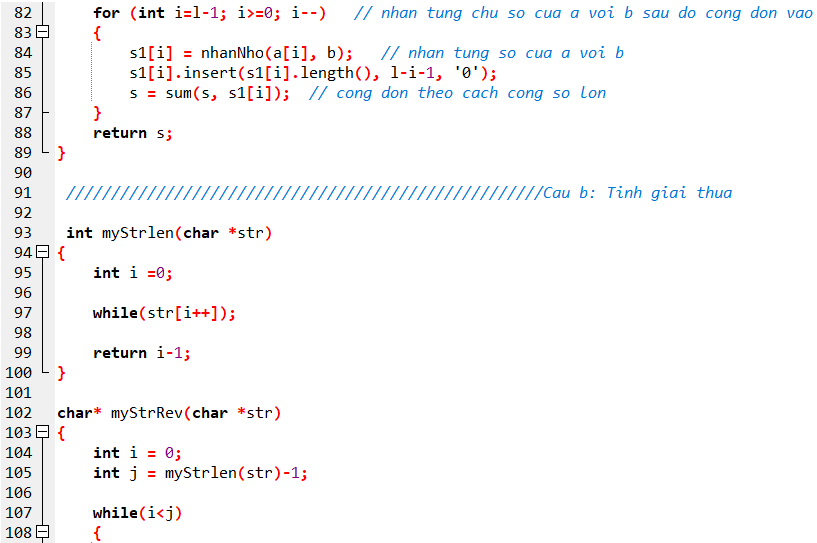
Trong bài này, thực hiện các phép toán +, \*, an, x!. Nhưng “**số nguyên**” để tính là số rất lớn, có số lượng ký số lên đến hàng chục, hàng ngàn và thậm chí là hàng tỉ ký số.

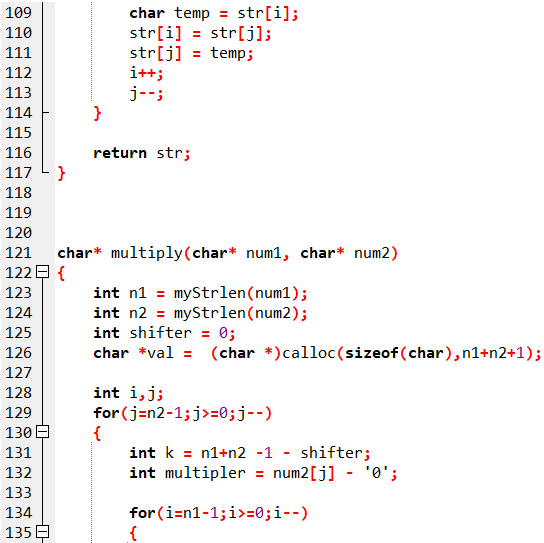
1. Cài đặt

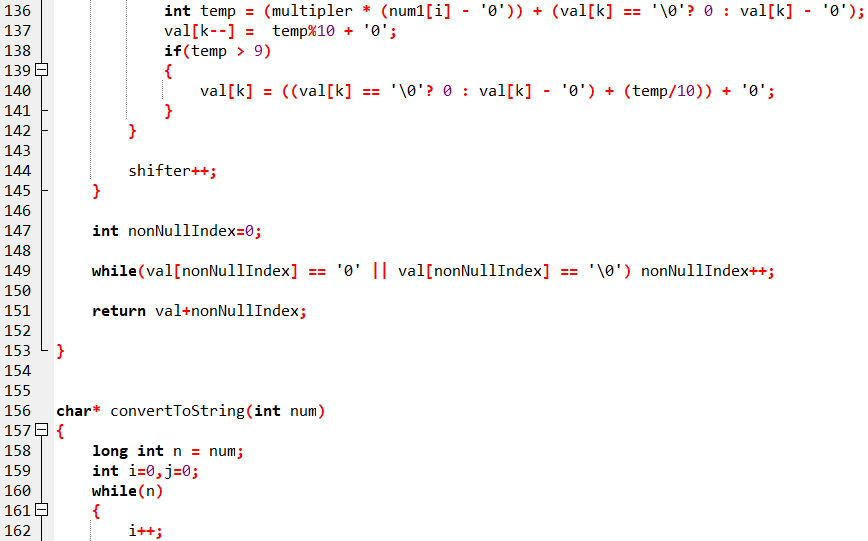


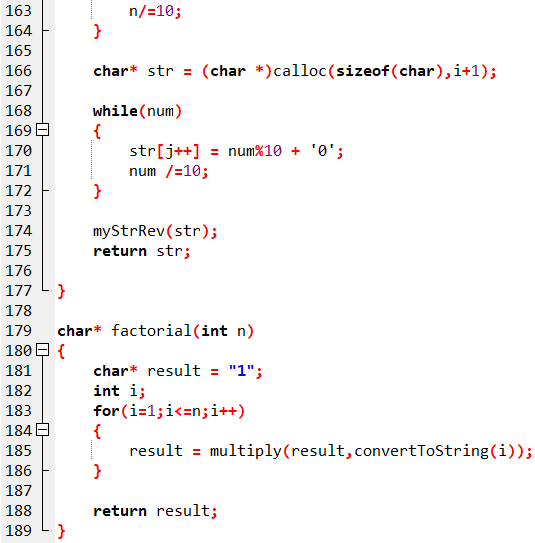


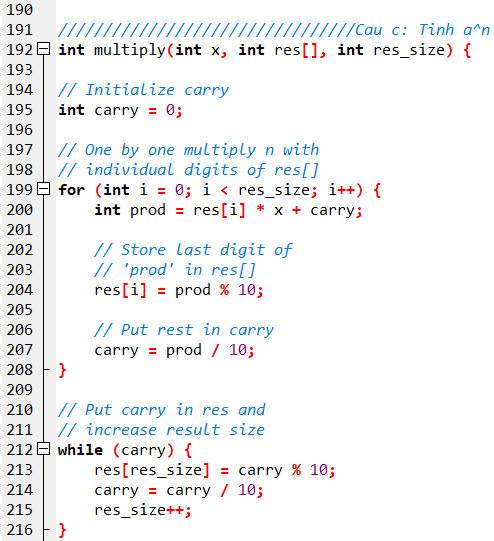


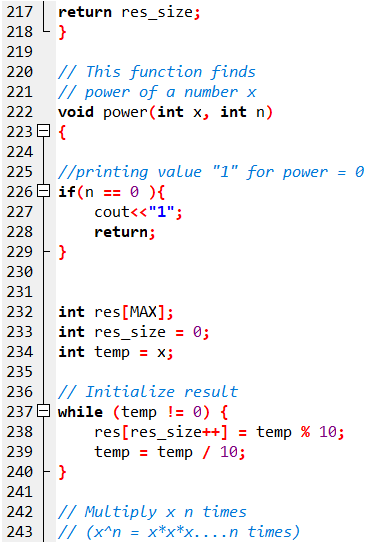


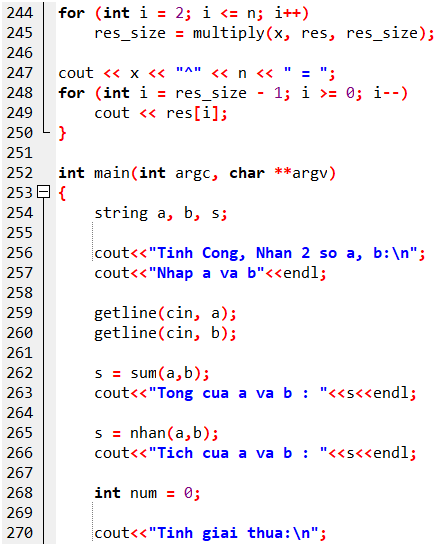


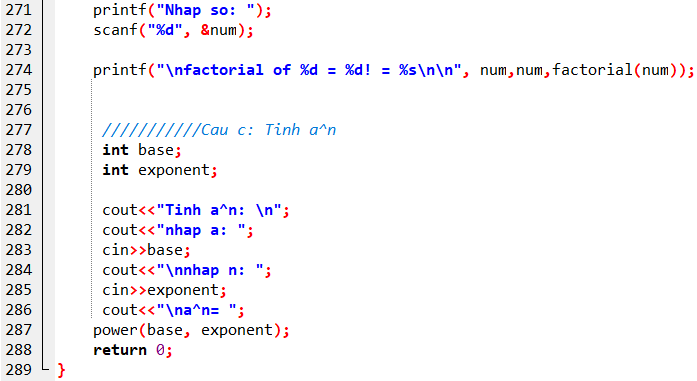




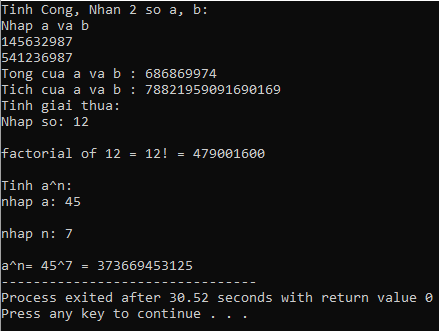








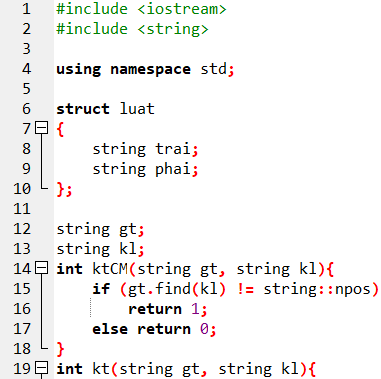
1. Kết quả

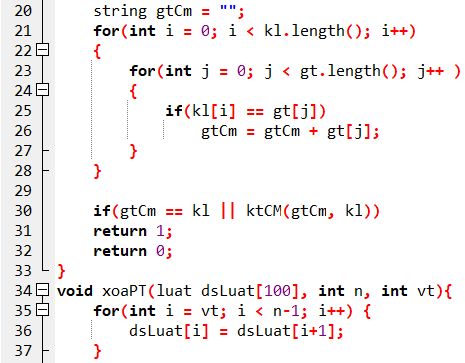


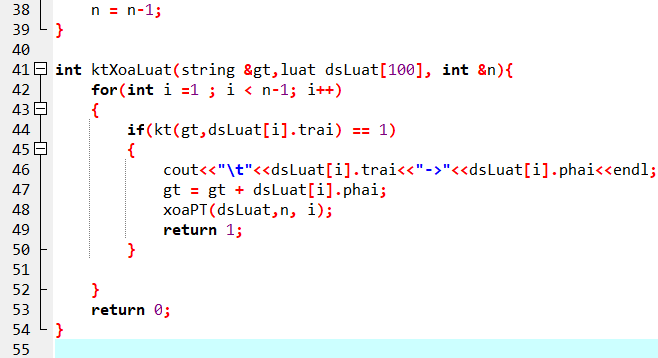
1. Cài đặt thuật toán Suy dẫn tiến
2. Cơ sở lý thuyết

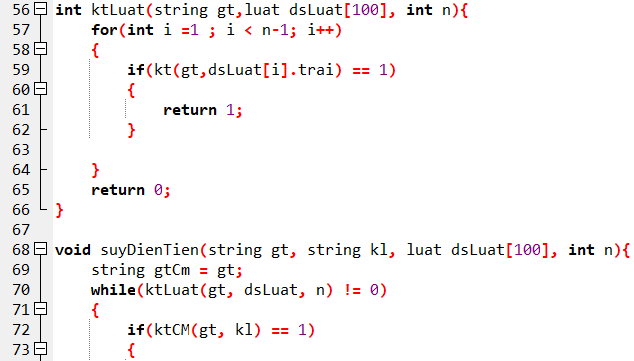
* Phương pháp: Suy dẫn từ giả thiết đi đến kết luận.
* Chiến lược này được bắt đầu bằng tập sự kiện đã biết, rút ra các sự kiện mới nhờ dùng các luật mà phần giả thiết khớp với sự kiện đã biết, và tiếp tục quá trình này cho đến khi thấy trạng thái đích, hoặc cho đến khi không còn luật nào khớp được các sự kiện đã biết hay được sự kiện suy luận.
* Trong áp dụng cụ thể phương pháp thường sử dụng kết hợp với các qui tắc Heuristic trong việc chọn luật.

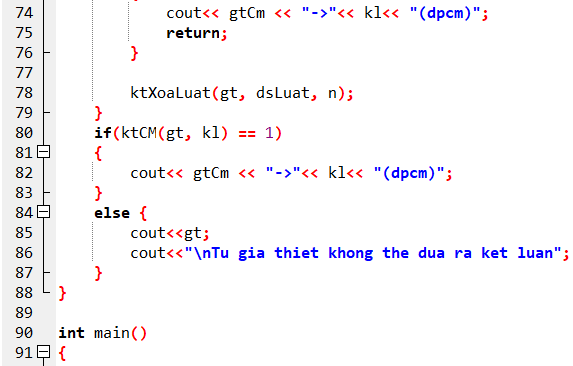
1. Cài đặt chương trình



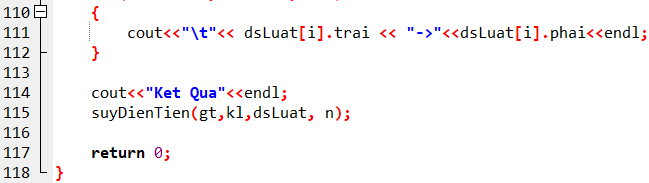






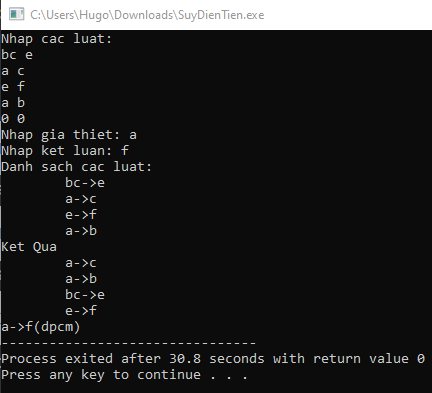




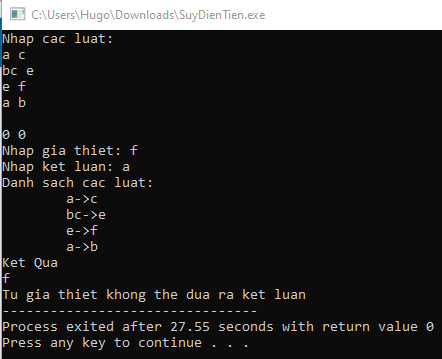


1. Kết quả

Chứng minh được



Không chứng minh được

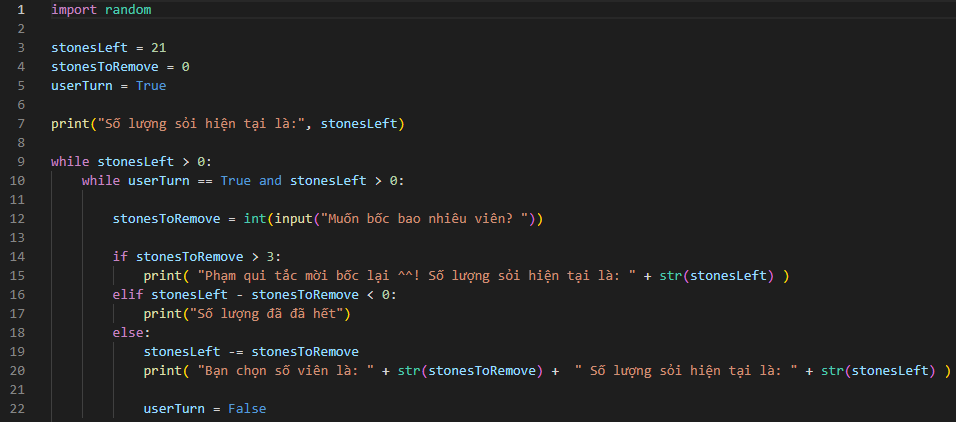


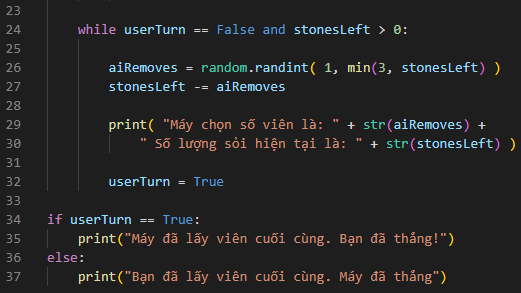
1. Cài đặt game Bóc sỏi

1. Cơ sở lý thuyết

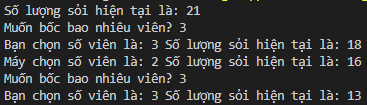
* **Bóc sỏi** là một trò chơi mà các người chơi tham gia phải bốc số lượng sỏi nhất định từ một đến ba viên người nào bốc viên cuối cùng thì người đó sẽ thua.

2. Cài đặt chương trình





3. Kết quả



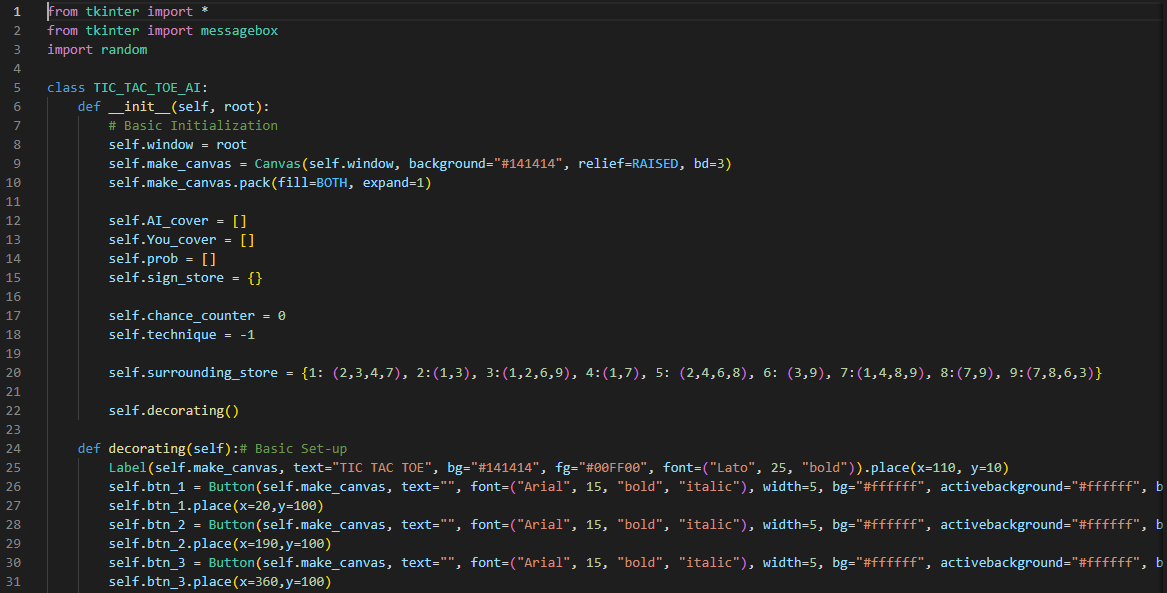


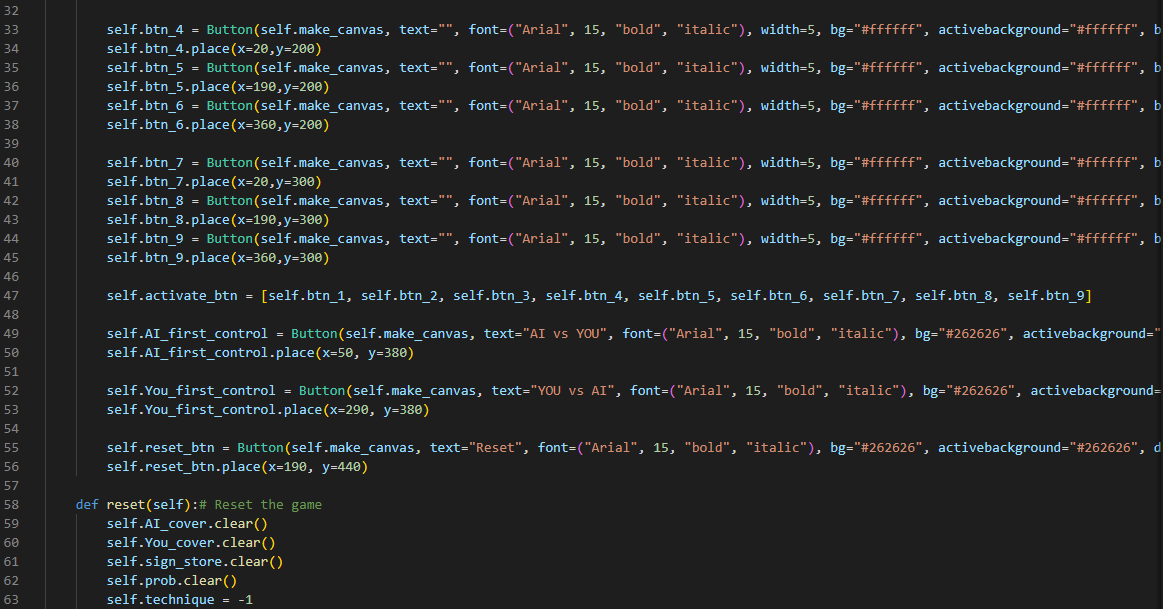
1. Cài đặt game Tic-tac-toe

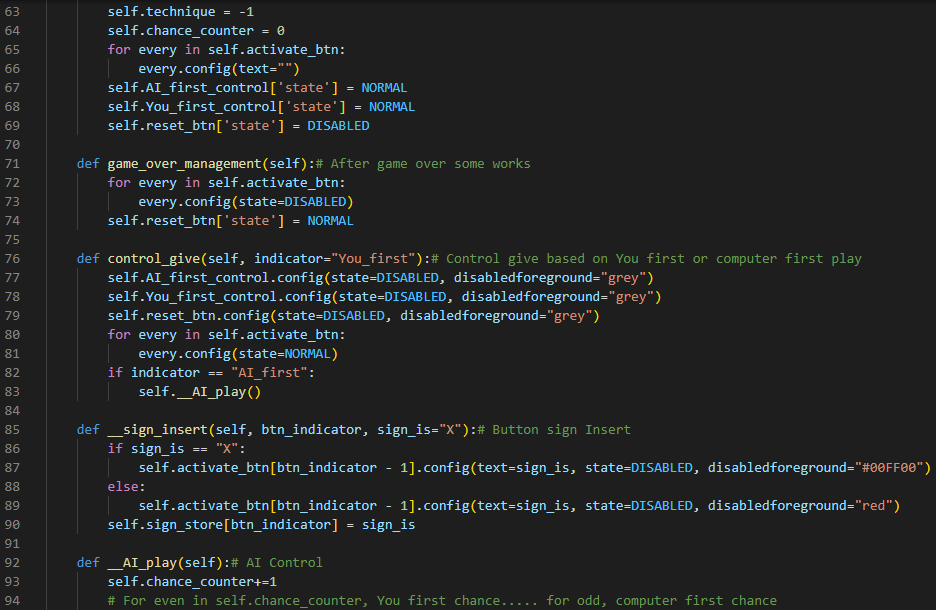
1. Cơ sở lý thuyết

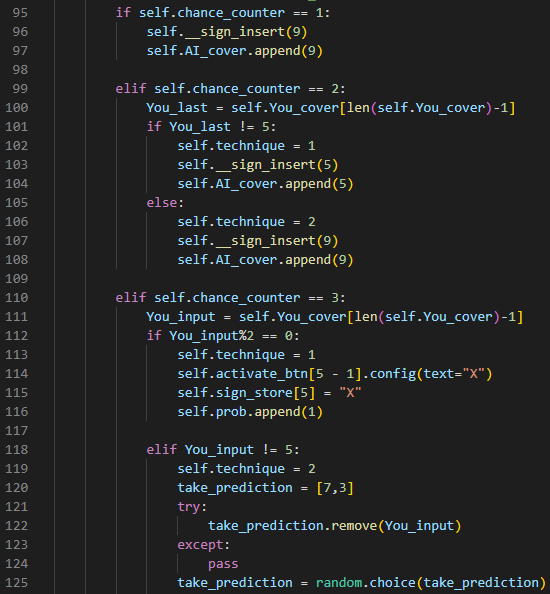
* **Tic-tac-toe** là một trò chơi phổ biến dùng viết trên bàn cờ giấy có chín ô, 3x3. Hai người chơi, người dùng ký hiệu O, người kia dùng ký hiệu X, lần lượt điền ký hiệu của mình vào các ô. Người thắng là người thể tạo được đầu tiên một dãy ty ký hiệu của mình, ngang dọc hay chéo đều được.
* Người Việt thường chơi trò tương tự, gọi là **Cờ carô**, bàn cờ không giới hạn trong 9 ô, có thể vẽ thêm ô, để kéo rộng ra cho đến khi người nào đạt được một dãy 5 thì thắng cuộc.

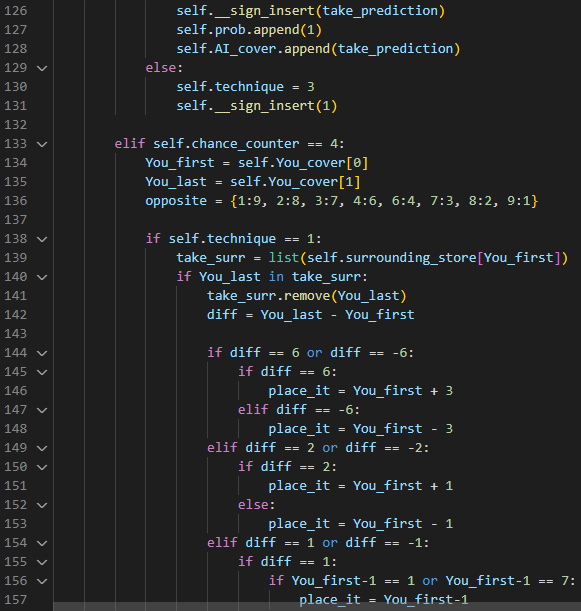
2. Cài đặt chương trình

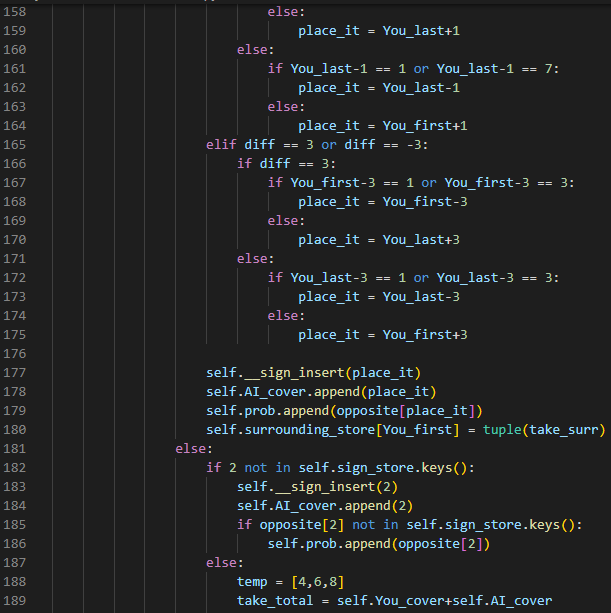


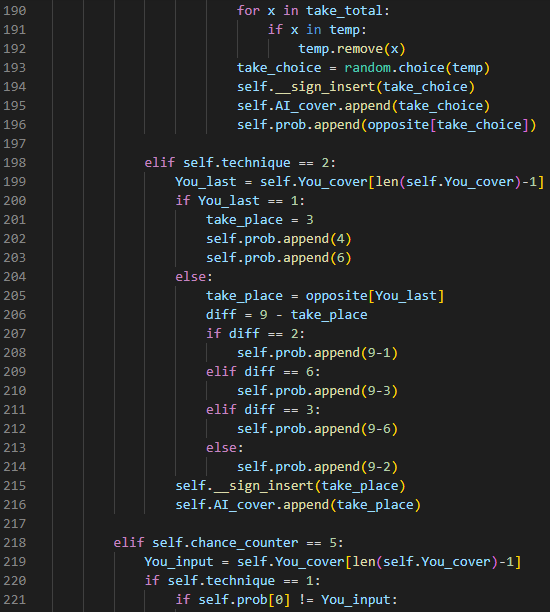






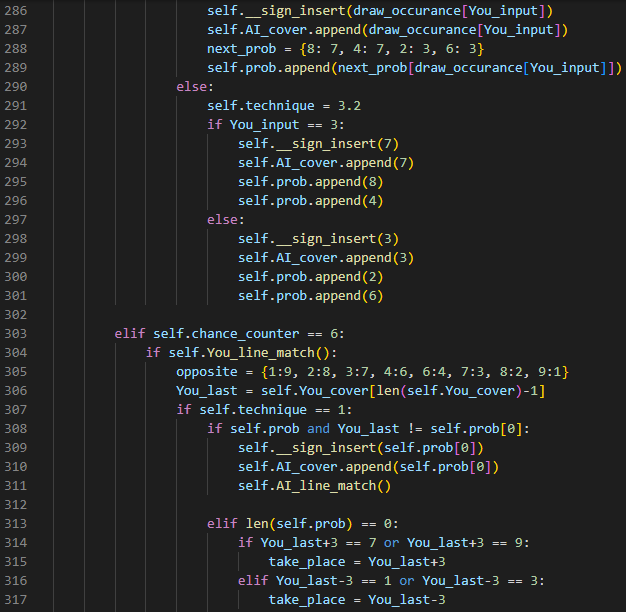


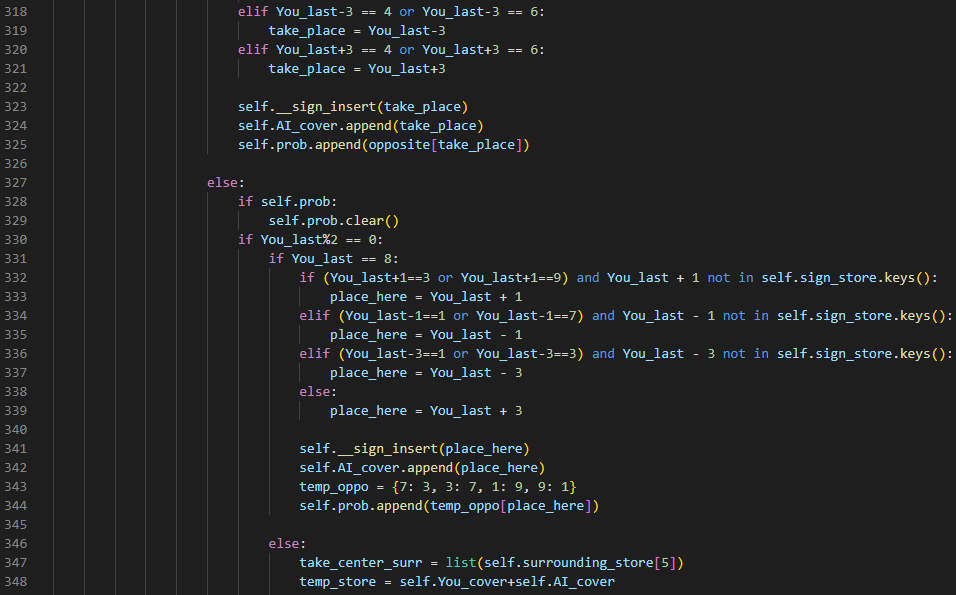


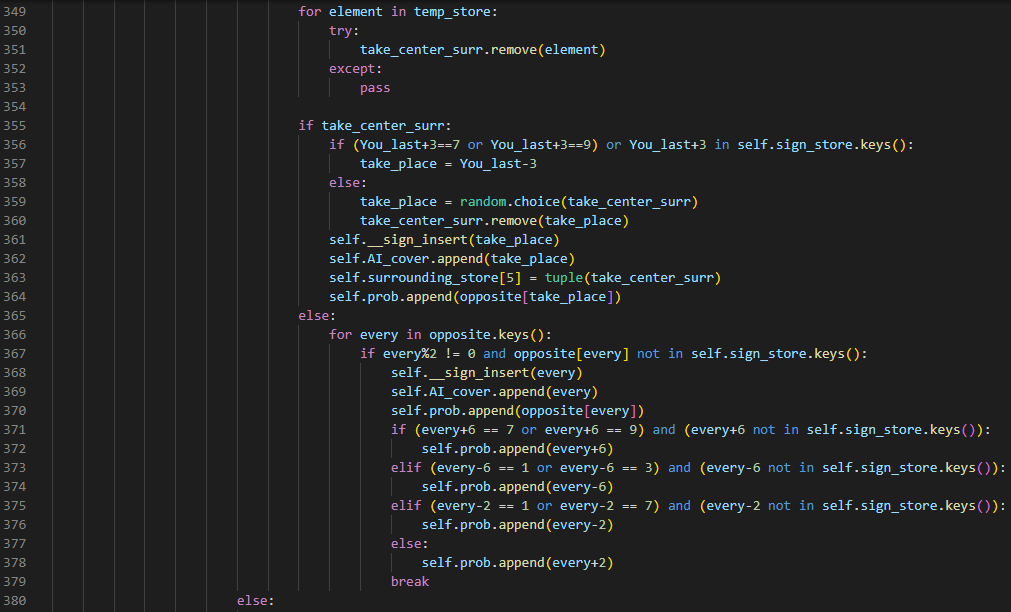


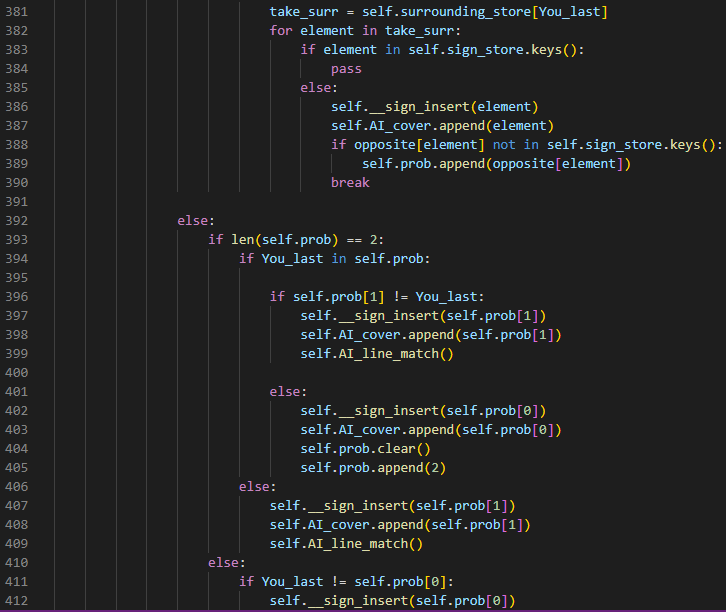


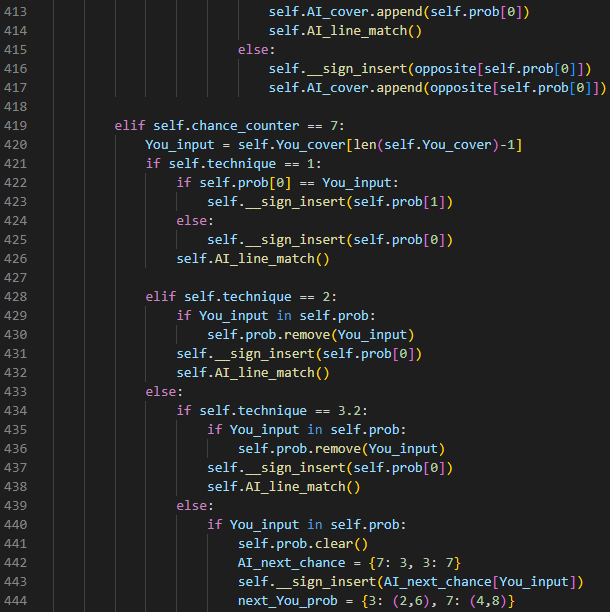


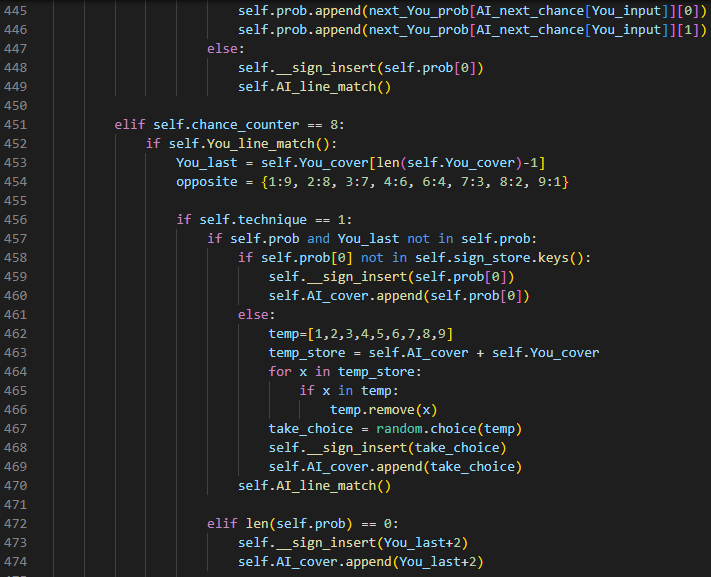




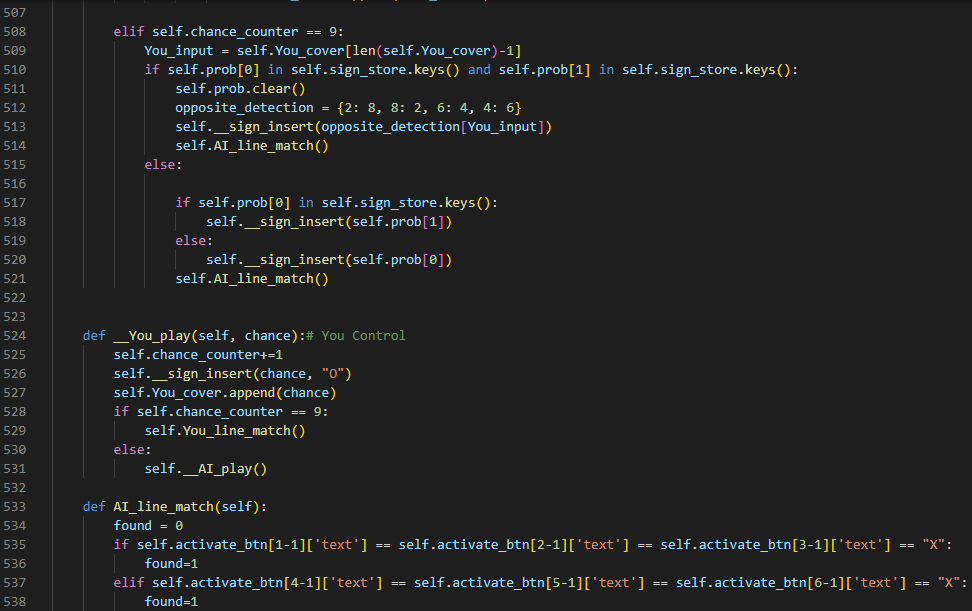


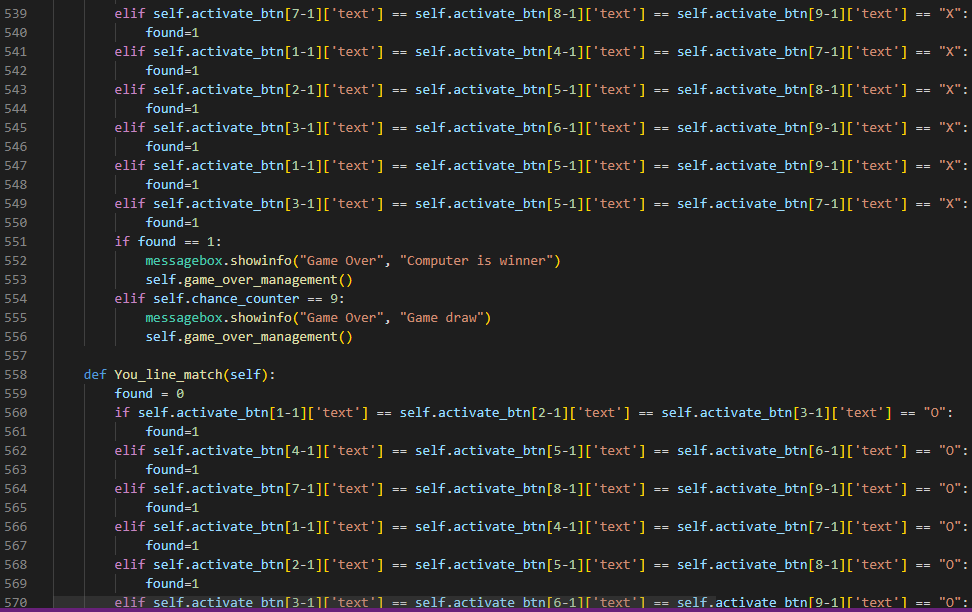














3. Kết quả



1. IoT – đồng hồ thời gian thực
2. Cở sở lý thuyết

Mô phỏng nguyên lý hoạt động bằng phần mềm Proteus

Việc mô phỏng nguyên lý của mạch có ý nghĩa hết sức quan trọng, thông qua việc mô phỏng các giải thuật được kiểm tra chi tiết nhằm tránh những sai sót không đáng có.

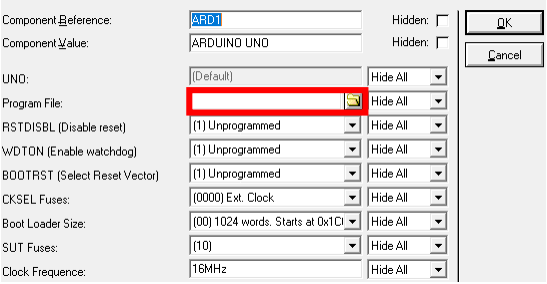


Trong hình, 13 LED 7 đoạn loại Cathode chung được dùng để hiển thị các thành phần của thời gian như: Thứ, Ngày, Tháng, Năm, Giờ, Phút, Giây. Thuật toán quét LED được áp dụng để Bật/Tắt các LED theo tuần tự với chu kỳ 1s cho mỗi lần hiển thị. Các chân BI/RBO được dùng cho giải thuật này. 12 IC 74LS47 (kết nối trực tiếp với LED 7 đoạn) được sử dụng để giải mã BCD sang LED 7 đoạn. Vì mỗi tín hiệu đầu ra của 74LS47cần được làm mới sau mỗi chu kỳ nên cần sử dụng đến IC74LS154 (IC ở vị trí gần với Arduino) để lần lượt quét qua các địa chỉ cho IC 74LS47. Do đầu ra của bộ giải mã địa chỉ luôn có mức tích cực thấp, IC 74LS04 (cổng logic NOT) được sử dụng để đảo ngược các tín hiệu này (2 IC góc trên cùng ở giữa). Module thời gian thực (khung màu đỏ) được dùng cho mô phỏng thời gian của đồng hồ. Board Arduino (màu xanh) là module dùng để lập trình mô phỏng. Các bước tiến hành mô phỏng như sau:

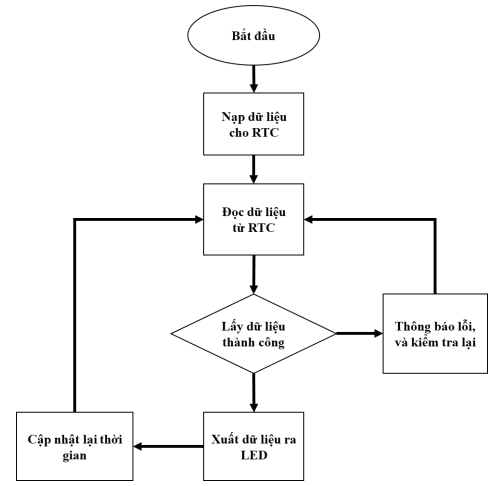
**Bước 1:** Hoàn thiện việc lấy linh kiện và nối dây liên kết

**Bước 2:** Lập trình cho kit Arduino bằng phần mềm Arduino IDE

**Bước 3:** Nạp code cho Arduino bằng cách D\_Click lên biểu tượng board lập trình và lựa chọn tập tin có phần mở rộng .HEX để thực thi. **Bước 4:** Nhấn nút  để bắt đầu mô phỏng, nhấn nút  để tạm dừng.

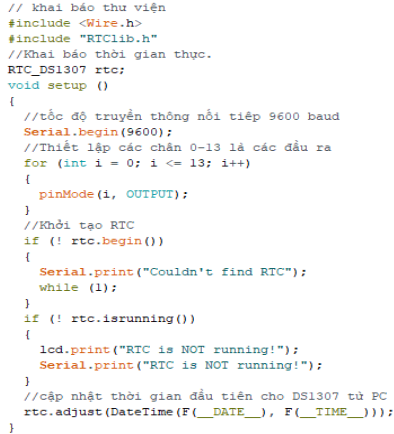


1. Cài đặt chương trình



**Giao tiếp RTC với Arduino**

Để cho RTC giao tiếp được với Arduino thì cần phải sử dụng đến thư viện “RTClib.cpp” và “RTClib.h” công dụng của thư viện này sẽ giúp cho việc lấy dữ liệu từ bộ nhớ của RTC ra xử lý và nạp vào vi điều khiển.



**Bước 1:** Khởi tạo RTC: *rtc.begin()*;

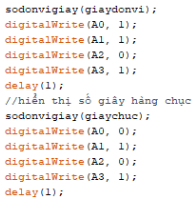
**Bước 2:** Kiểm tra trạng thái hoạt động của RTC: *!rtc.isrunning()*, kết quả trả về giá trị TRUE khi RTC chưa hoạt động, ngược lại là FALSE.

**Bước 3:** Cập nhập thời gian cho RTC: *rtc.adjust(DateTime(F(\_DATE\_),F(\_TIME\_)))*;

**Hiển thị thời gian lên Led 7 đoạn**

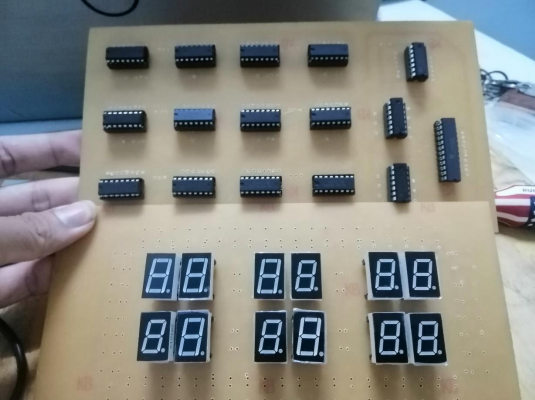
Để hiển thị thời gian lên LED 7 đoạn cần sử dụng đến kĩ thuật quét LED, với tất cả các chân DATA của IC 74LS47 được nối chung vói nhau đồng nghĩa với việc tất cả LED 7 đoạn đều sáng khi dữ liệu được gửi lên 4 bit data. Tuy nhiên khi chân BI/RBO và chân RBI ở mức thấp thì LED sẽ không hiển thị, như vậy kĩ thuật quét LED sẽ là hiển thị từng LED tương ứng với BI/RBO và RBI của IC 74LS47 tương ứng với giá trị là 1, sau đó tắt LED này đi với giá trị BI/RBO và RI tương ứng với giá trị bằng 0. Tiếp tục thực hiện cho các LED còn lại.

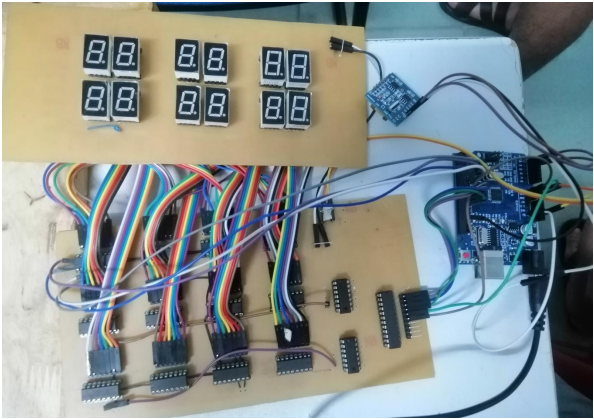
Các bước hiển thị số thời gian giây hàng đơn vị và hàng chục với giải thuật quét LED. Trong đó, 4 bit dữ liệu A0 – A3 là địa chỉ của BI/RBO và RI tương ứng với việc hiển thị LED giây hàng đơn vị và LED giây hàng chục. Mỗi LED được thắp sáng trong khoảng thời gian 1 chu kỳ máy (hàm delay (1)), tương tự như các thời gian năm, tháng, ngày, giờ, phút (xem mã nguồn ở bảng phụ lục).



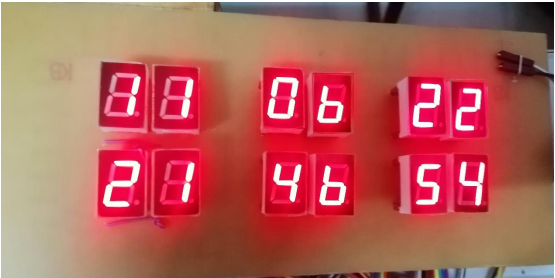
1. Kết quả

Kết quả thiết kế gia công mạch in hoành thành sẽ được mạch điện với các LED 7 đoạn và các IC được hàn cố định. Trong đó, các IC số cho việc điều khiển hiển thị và quét LED và phần LED 7 đoạn hiển thị.



Sau khi tiến hành nối dây cho các thành phần, kết quả như hình. 

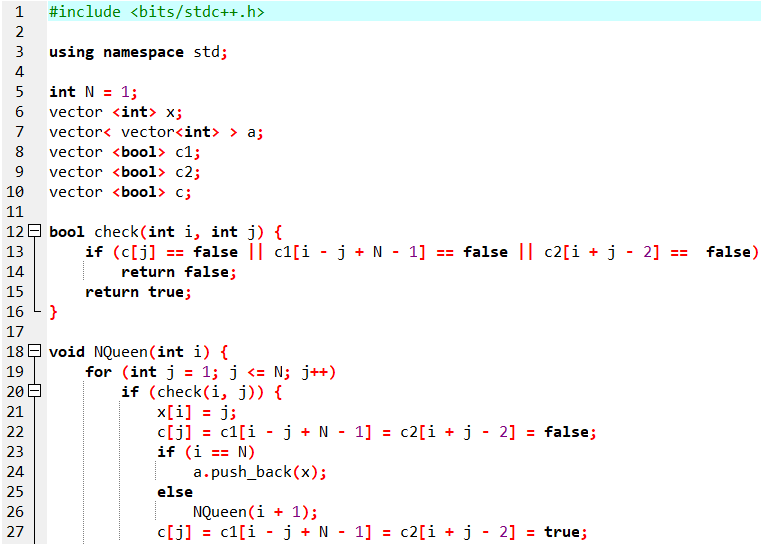
Chương trình được nạp vào board mạch Arduino, thời gian được hiển thị lên LED 7 đoạn ở theo từ trái sáng phải từ trên xuống dưới tương ứng với ngày, tháng năm, giờ, phút, giây. Như hình đồng hồ thời gian thực tương ứng với ngày 11 tháng 06 năm 2022, 21 giờ 46 phút 54 giây.

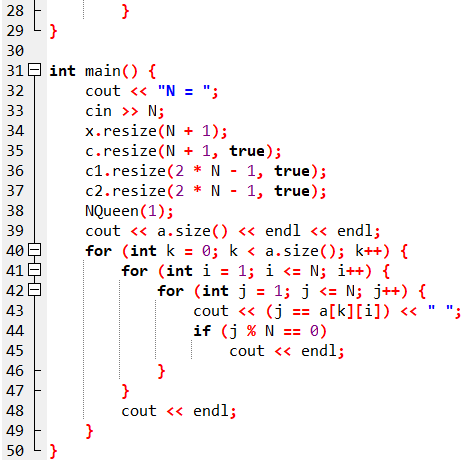


1. Cài đặt bài toán 8 quân hậu
2. Cở sở lý thuyết

* Bàn cờ là một bàn vuông 8x8, nên ta dùng một mảng 2 chiều với kích thước [8, 8] để tổ chức bàn cờ.
* Các ô nằm theo chiều ngang là các dòng, các ô nằm dọc gọi là các cột.
* Đường chéo chính là đường chéo mà các hiệu số dòng và cột bằng nhau.
* Đường chéo phụ là đường chéo mà các tổng số dòng và cột bằng nhau.
* Đường chéo Trừ là đường chéo song song với đường chéo chính.
* Đường chéo Cộng là đường chéo song song với đường chéo phụ.

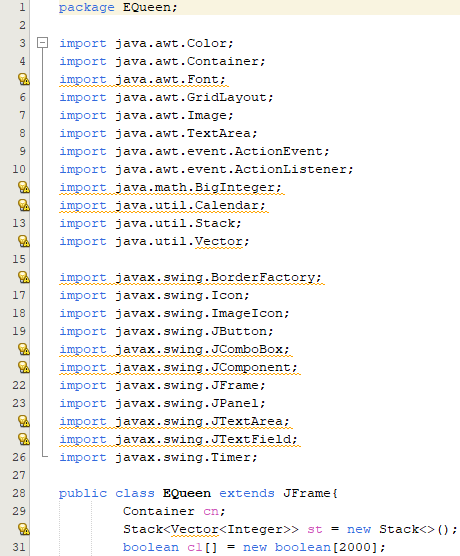
1. Cài đặt chương trình

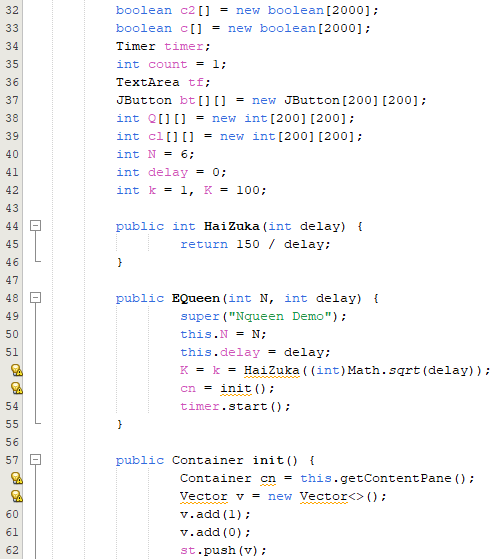


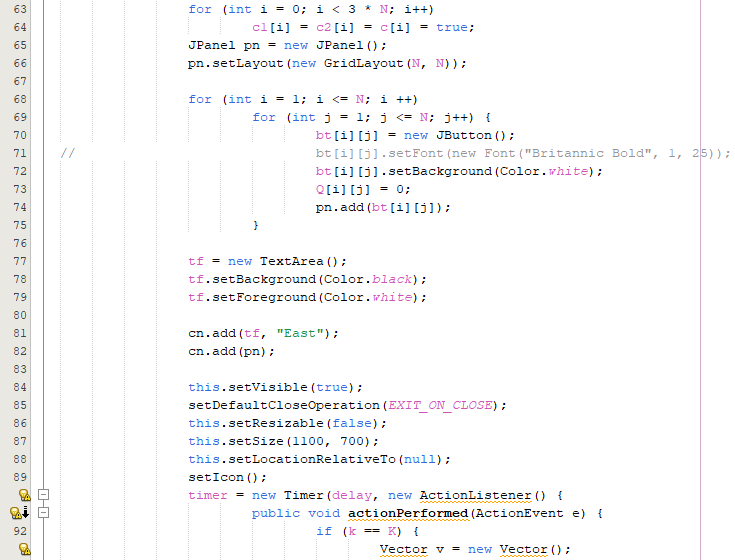


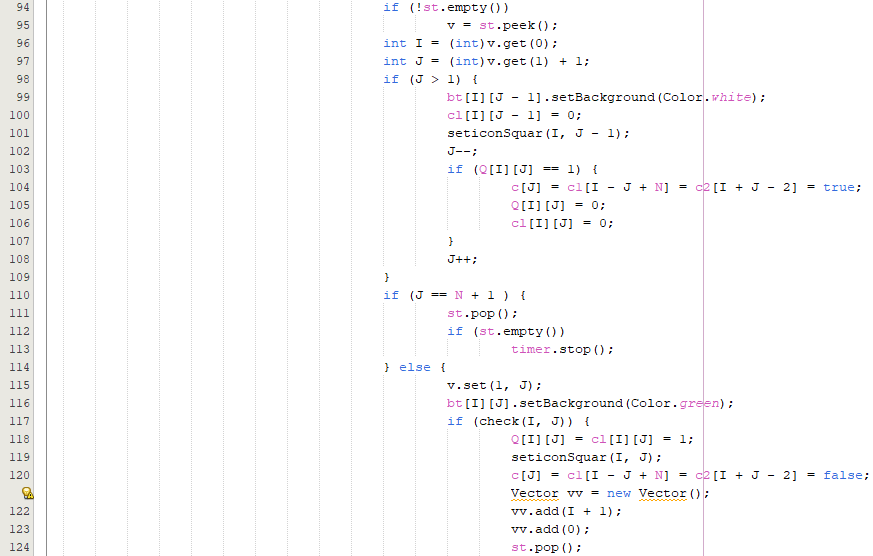
Code demo

Class Equeen

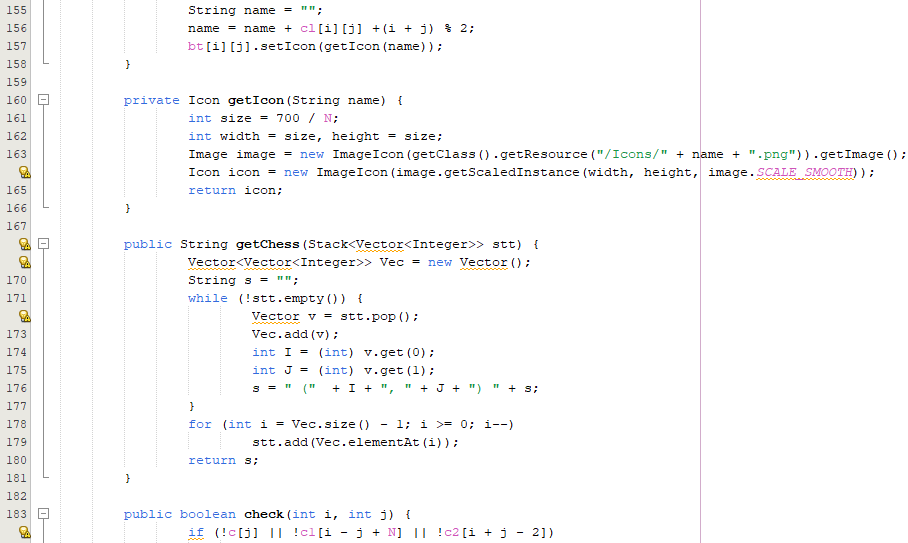


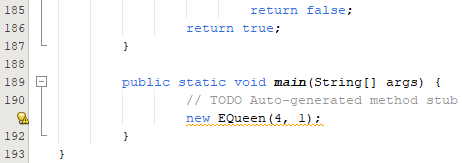




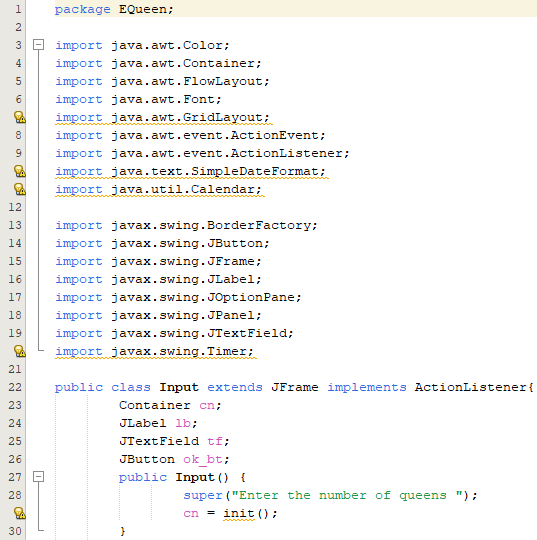








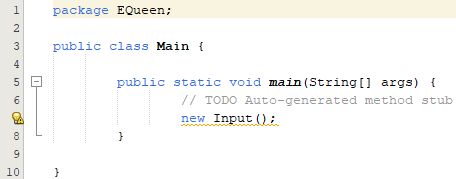
Class Input



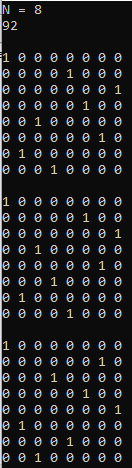




Class Main



3. Kết quả



1. Cài đặt thuật toán Robinson

1. Cơ sở lý thuyết

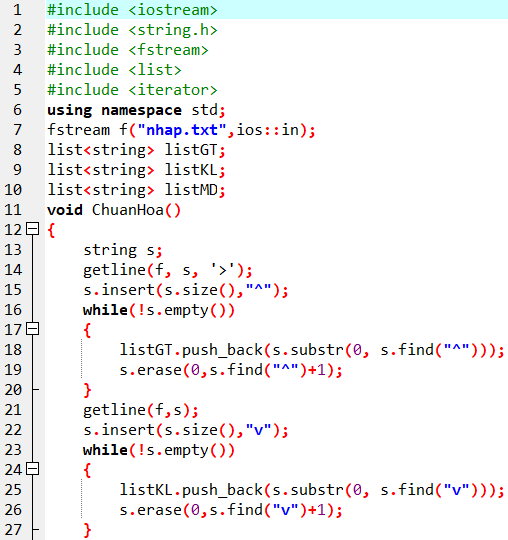
* Bước 1: Phát biểu lại giả thuyết và kết luận bài toán dưới dạng chuẩn sau:

GT1, GT2, …, GTn -> KL1, KL2, …, KLm

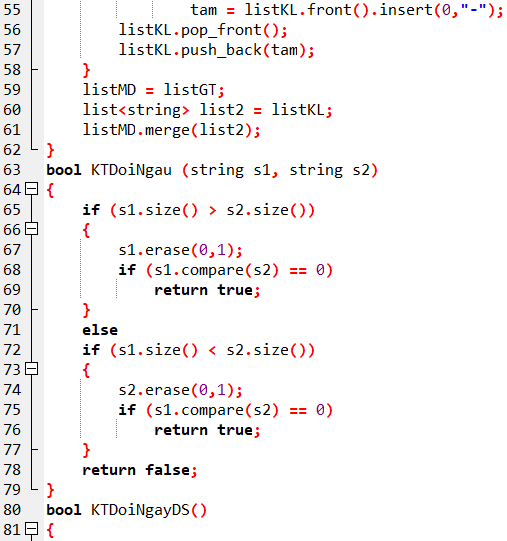
Trong đó các GTi và KLj được xây dựng từ các biến mệnh đề và các phép toán: http://sinhvientot.net/wp-content/uploads/2017/05/image013.png

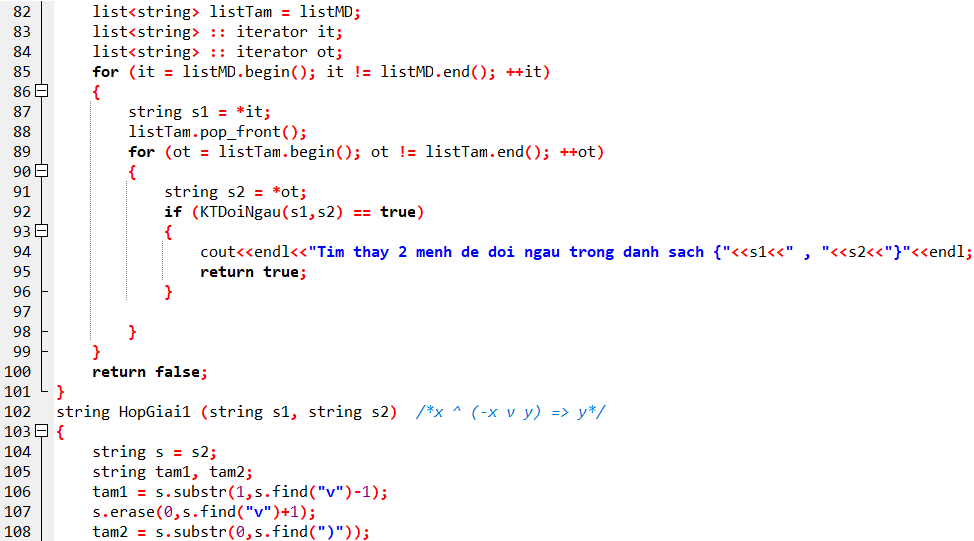
* Bước 2: Biến đổi dòng trên thành danh sách các mệnh đề
* Bước 3: Nếu trong danh sách các mệnh đề ở bước 2 có hai mệnh đề đối ngẫu nhau thì vấn đề được giải quyết xong, còn không chuyển sang bước 4 (dạng {a,a}).
* Bước 4: Xây dựng một mệnh đề mới bằng cách tuyển một cặp mệnh đề trong danh sách các mệnh đề ở bước 2. Nếu mệnh đề mới có các biến mệnh đề đối ngẫu thì những biến đó được loại bỏ.
* Bước 5: Bổ sung mệnh đề mới vào danh sách và loại bỏ hai mệnh đề cũ vừa tạo thành mệnh đề mới ra khỏi danh sách.
* Bước 6: Nếu không xây dựng thêm mệnh đề mới nào vào trong danh sách các mệnh đề không có hai mệnh đề nào đối ngẫu nhau thì vấn đề phát biểu ở dạng chuẩn bước 1 là sai.

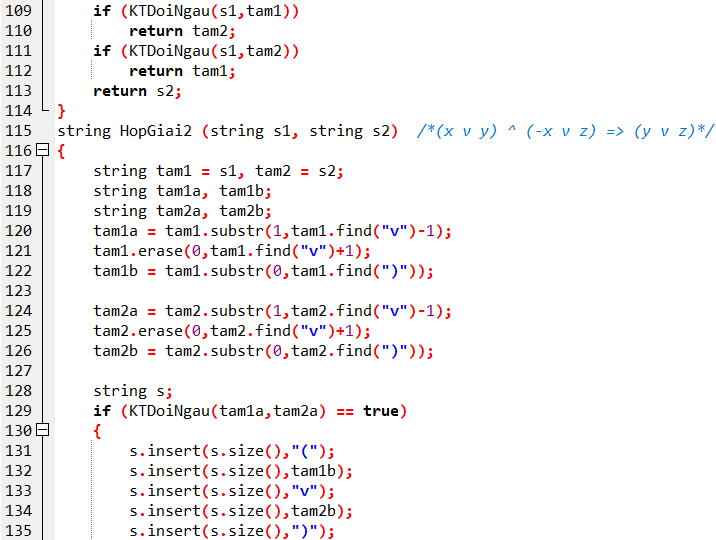
2. Cài đặt chương trình

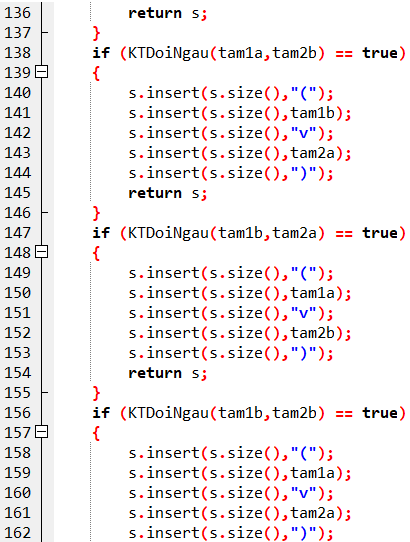


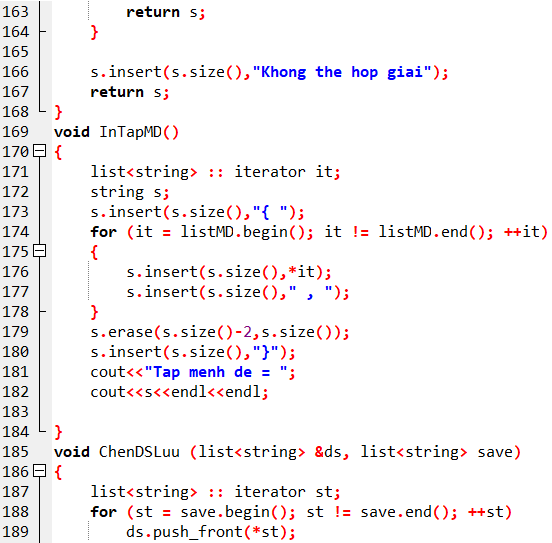


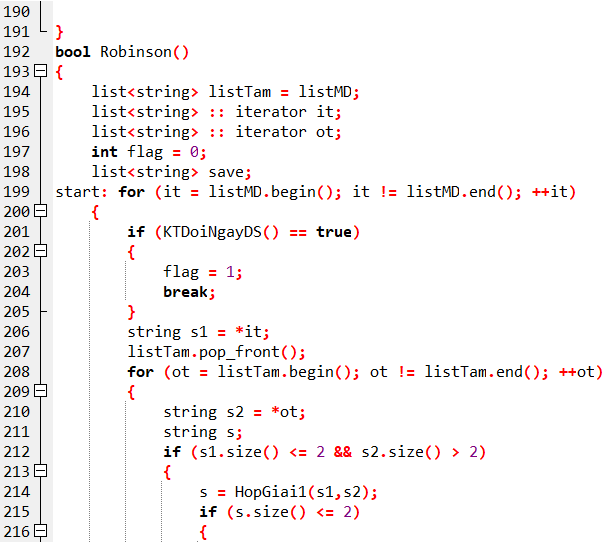


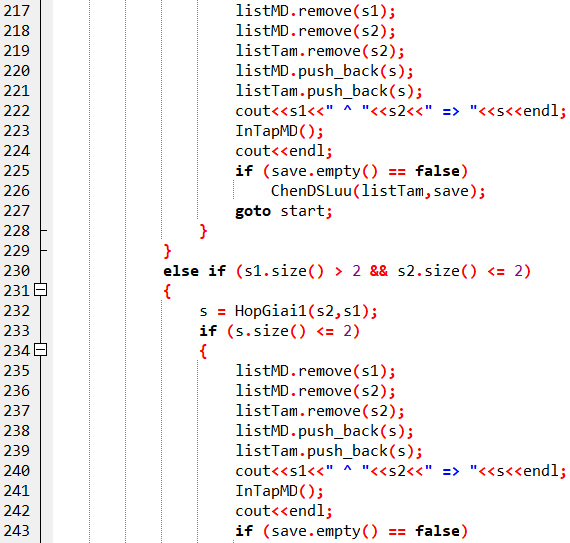


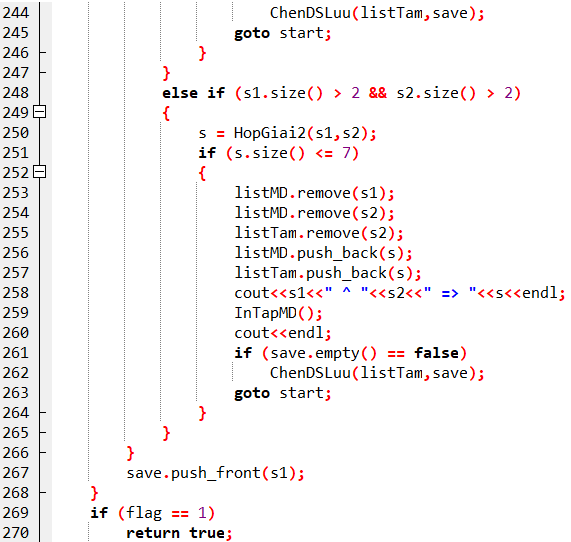






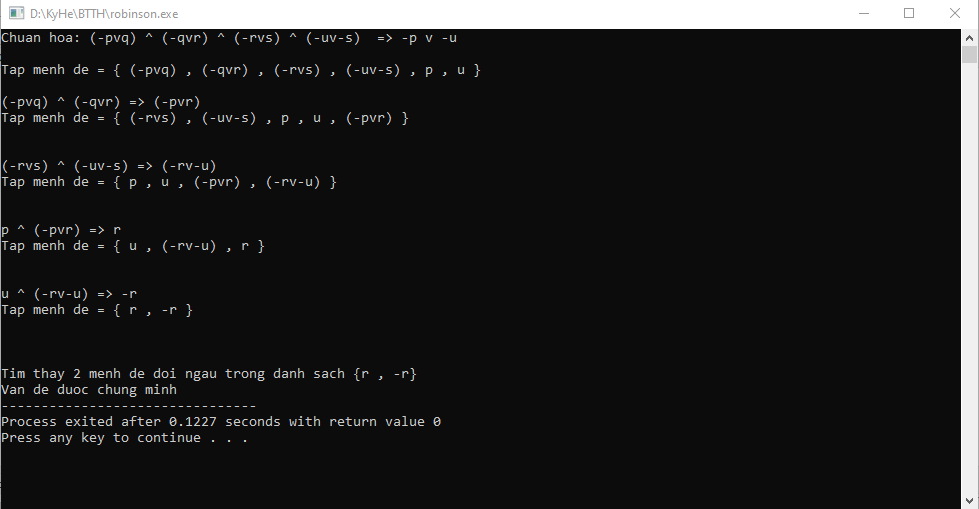








3. Kết quả



1. Cài đặt thuật toán Vương hạo

1. Cơ sở lý thuyết

* Bước 1: Phát biểu lại giả thuyết và kết luận của bài toán dưới dạng chuẩn sau:

GT1, GT2, …, GTn → KL1, KL2, … KLm

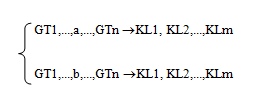
Trong đó các GTi và KLj được xây dựng từ các biến mệnh đề và các phép toán: “∧,∨,¬”

* Bước 2: Chuyển vế các GTi và KLj có dạng phủ định.
* Bước 3: Thay dấu “∧” ở trong GTi và dấu “∨” ở trong KLj bằng dấu “,”
* Bước 4: Nếu dòng hiện hành có một trong hai dạng sau:

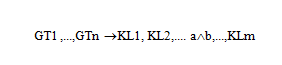
+ Dạng 1:

http://sinhvientot.net/wp-content/uploads/2017/05/1.png

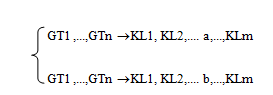
Thì thay bằng 2 dòng:



+ Dạng 2:



Thì thay bằng 2 dòng:

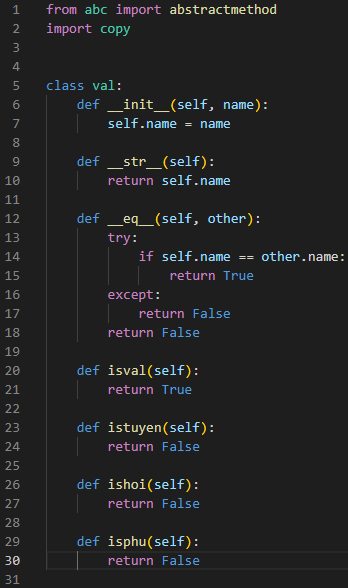


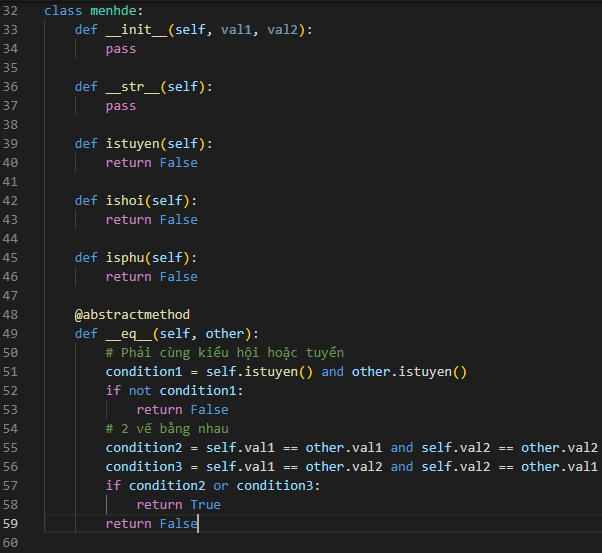
* Bước 5: Nếu một dòng được chứng minh: nếu tồn tại chung một mệnh đề ở cả 2 vế thì coi như đúng.
* Bước 6:

+ Nếu một dòng không còn dấu liên kết tuyển và hội mà cả ở hai vế đều không có chung biến mệnh đề nào thì dòng đó không được chứng minh.

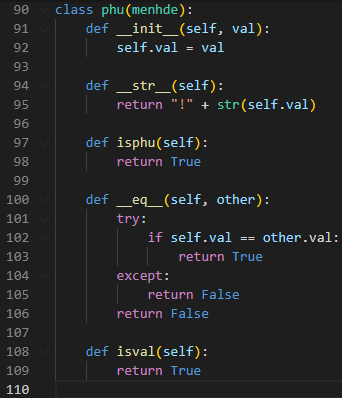
+ Một vấn đề được giải quyết một cách trọn vẹn nếu mọi dòng dẫn xuất từ dạng chuẩn được chứng minh.

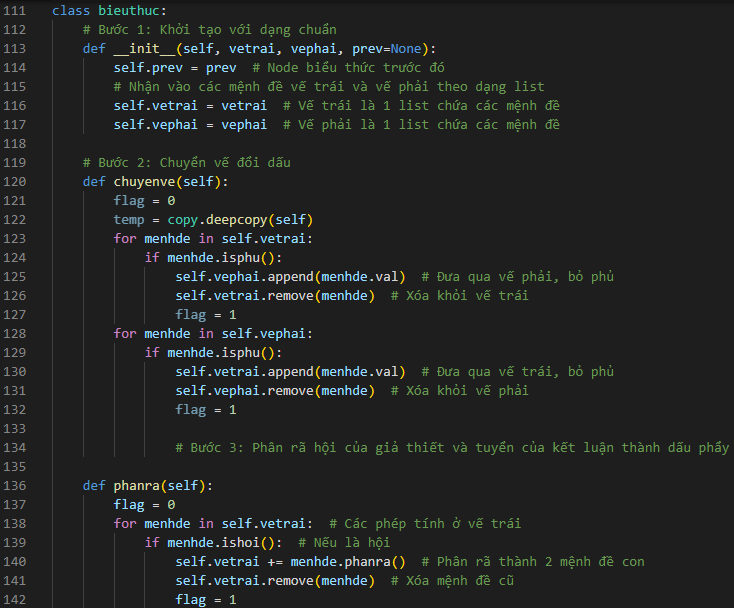
2. Cài đặt chương trình



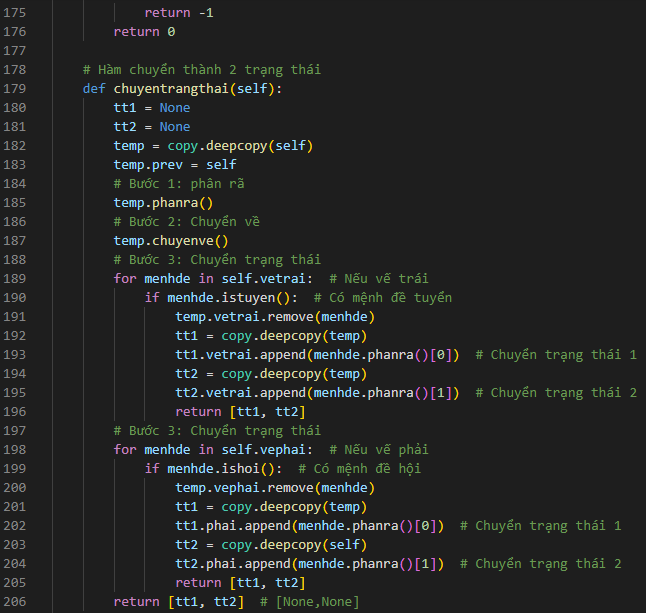


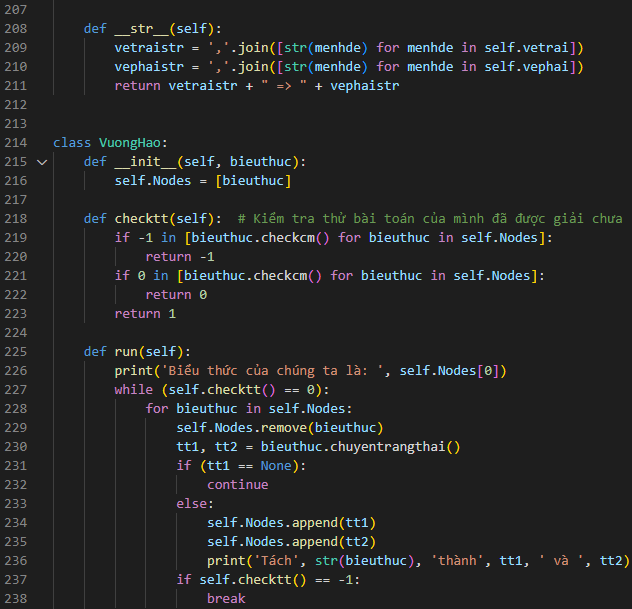


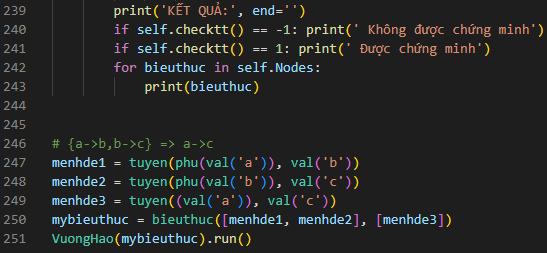




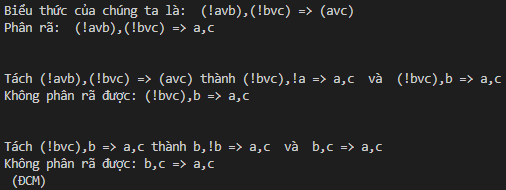


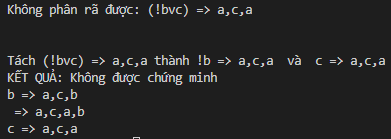






3. Kết quả

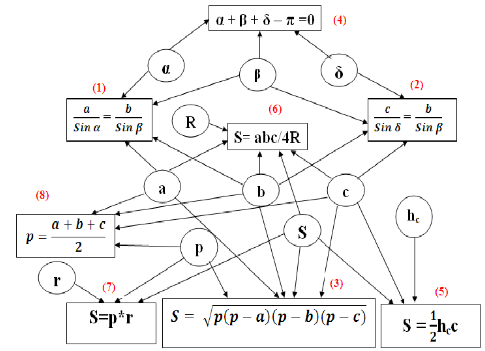




1. Cài đặt Bài toán tam giác

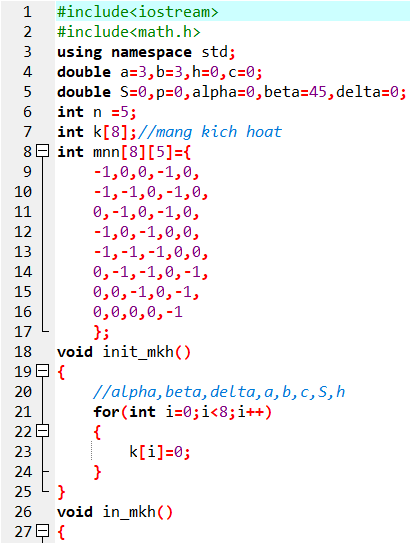
1. Cơ sở lý thuyết

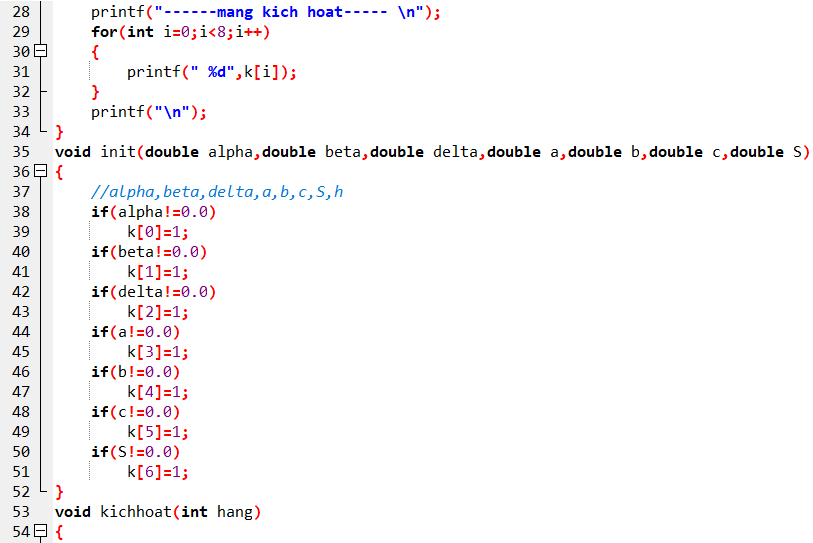
Ta có mạng ngữ nghĩa được xây dựng để giải bài toán tam giác như sau bao gồm 8 công thức và 11 yếu tố liên quan:

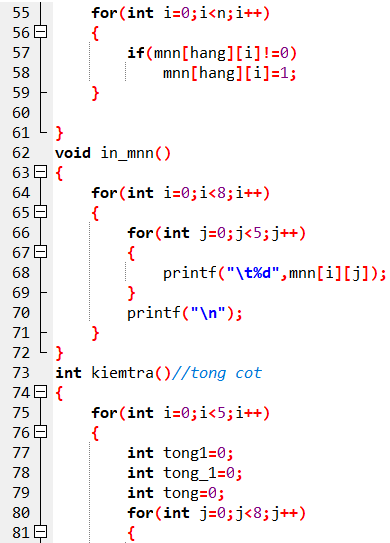


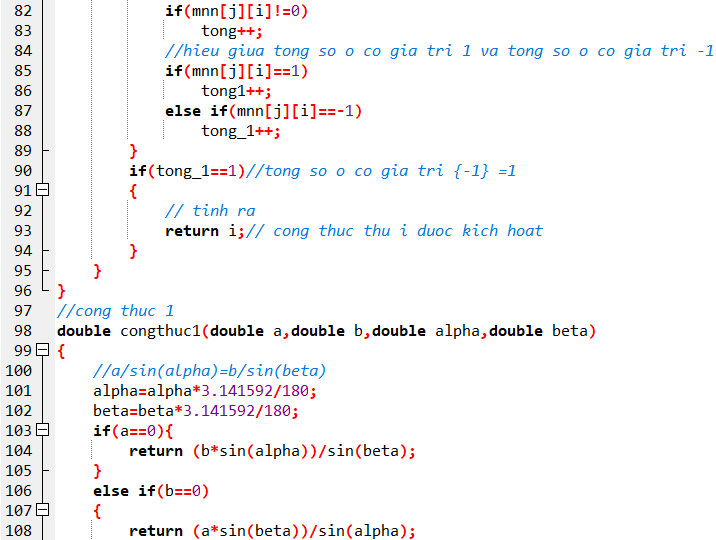
* 1. Các đỉnh α, β và a được kích hoạt
  2. Công thức (1) được kích hoạt (có ¾ yếu tố đã được kích hoạt). Từ công thức (1) tính được cạnh b -> đỉnh b được kích hoạt.
  3. Công thức (4) được kích hoạt (α, β đã được kích hoạt). Từ công thức (4) tính được góc c -> đỉnh δ được kích hoạt.
  4. Công thức (2) được kích hoạt (α, β, δ đã được kích hoạt). Từ công thức (2) tính được cạnh c -> đỉnh c được kích hoạt.
  5. Công thức (8) được kích hoạt (a, b, c đã được kích hoạt). Từ công thức (8) tính được nửa chu vi p -> đỉnh p được kích hoạt.
  6. Công thức (3) được kích hoạt (p, a, b, c đã được kích hoạt). Từ công thức (3) tính được diện tích S -> đỉnh S được kích hoạt.
  7. Công thức (5) được kích hoạt (S, c đã được kích hoạt). Từ công thức (5) tính được He -> đỉnh He được kích hoạt.
  8. Công thức (6) được kích hoạt (a, b, c, S đã được kích hoạt). Từ công thức (6) tính được bán kính R -> đỉnh R được kích hoạt.
  9. Giá trị R đã tính được -> Thuật toán kết thúc.

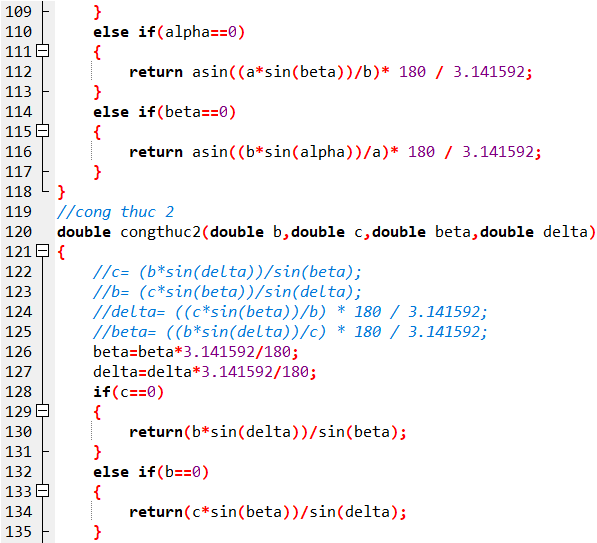
1. Cài đặt chương trình

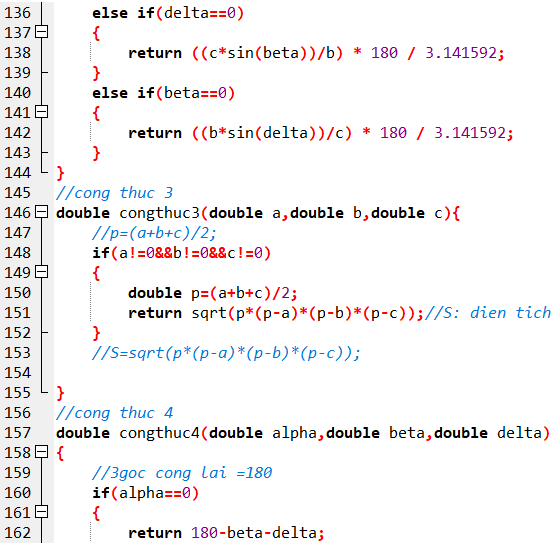


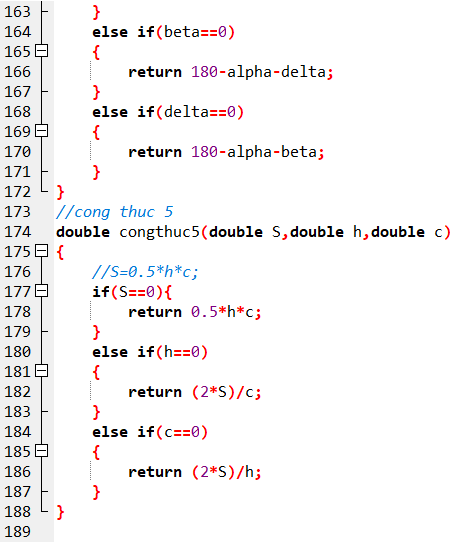




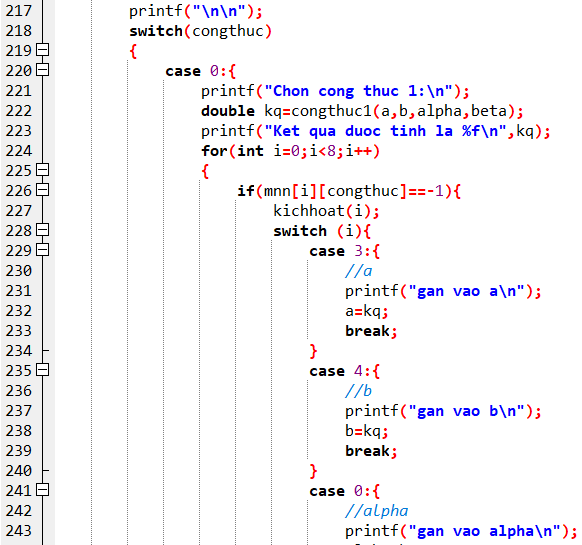


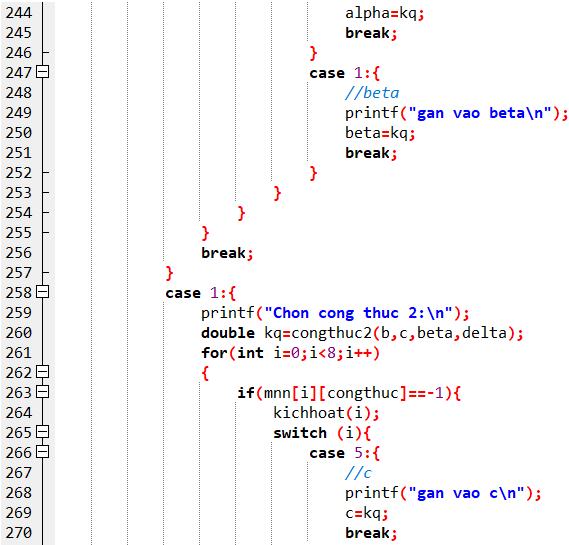


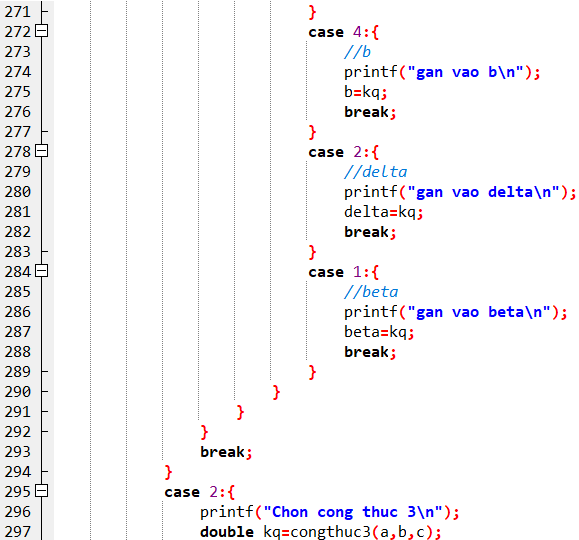


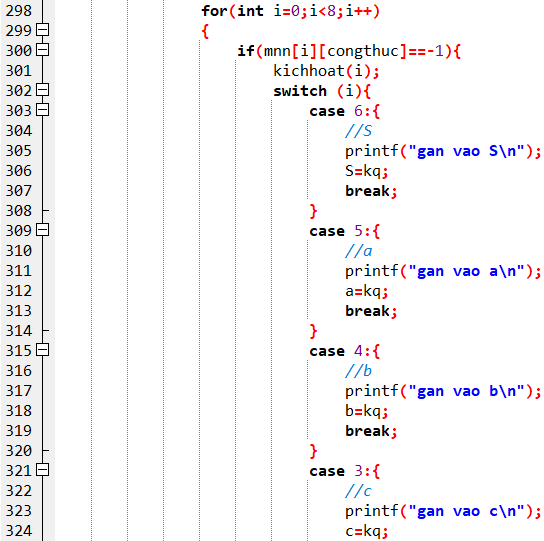


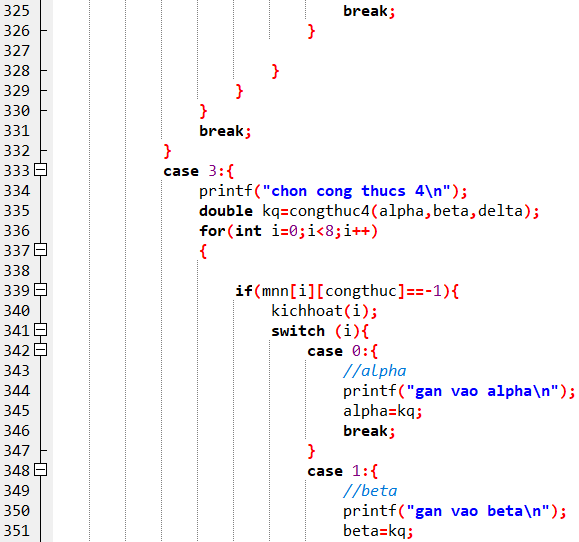


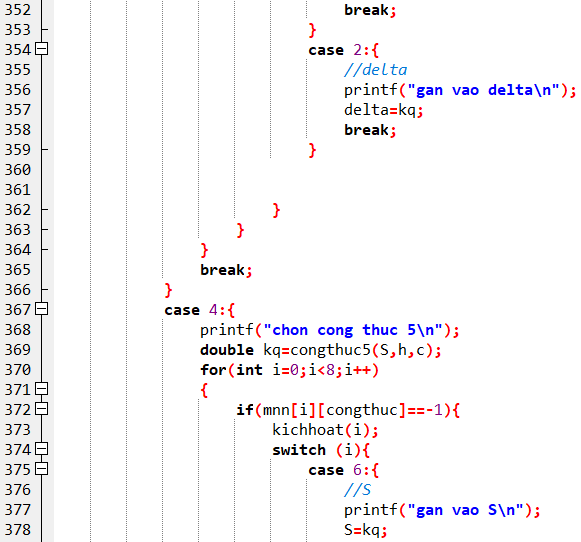


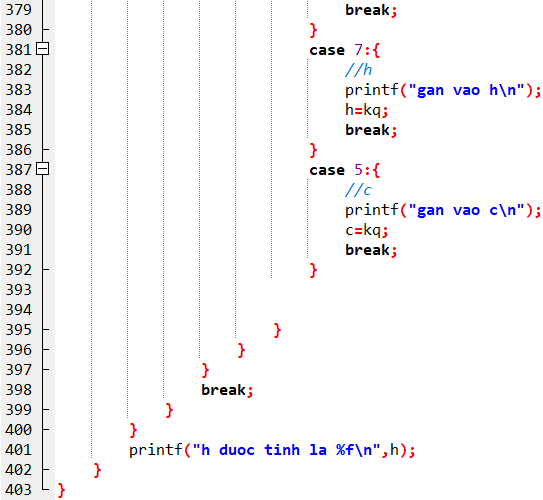




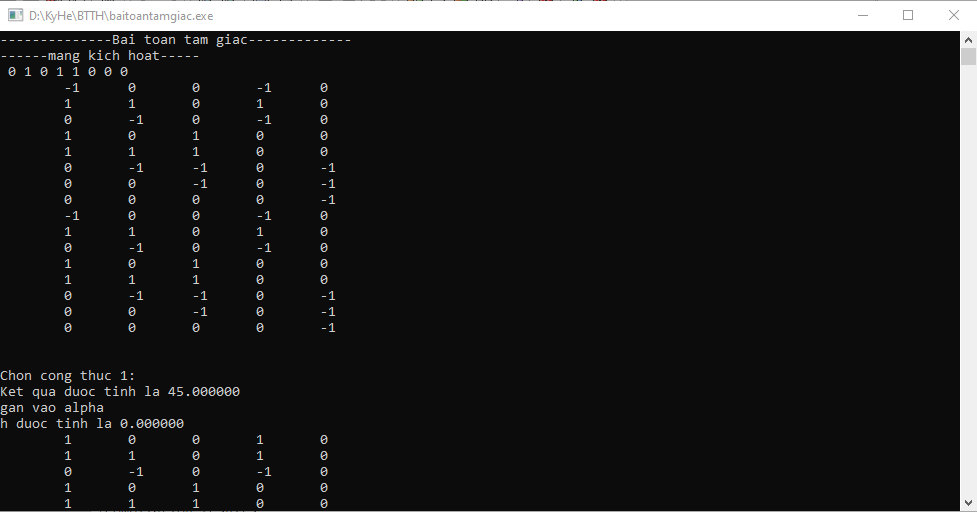








3. Kết quả



1. Cài đặt thuật toán di truyền
2. Cơ sở lý thuyết

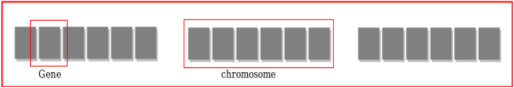
Thuật toán di truyền (GA) là thuật toán tìm kiếm theo phương pháp heuristic thích ứng thuộc về phần lớn hơn của thuật toán tiến hóa. Các thuật toán di truyền dựa trên ý tưởng của chọn lọc tự nhiên và di truyền. Đây là những khai thác thông minh của tìm kiếm ngẫu nhiên được cung cấp với dữ liệu lịch sử để hướng tìm kiếm vào vùng có hiệu suất tốt hơn trong không gian giải pháp. Chúng thường được sử dụng để tạo ra các giải pháp chất lượng cao cho các vấn đề tối ưu hóa và các vấn đề tìm kiếm.

Các thuật toán di truyền mô phỏng quá trình chọn lọc tự nhiên có nghĩa là những loài nào có thể thích nghi với những thay đổi của môi trường thì có thể tồn tại, sinh sản và chuyển sang thế hệ tiếp theo. Nói một cách đơn giản, chúng mô phỏng “sự sống sót của những người khỏe nhất” giữa những người thuộc thế hệ liên tiếp để giải quyết một vấn đề. Mỗi thế hệ bao gồm một quần thể các cá thể và mỗi cá nhân đại diện cho một điểm trong không gian tìm kiếm và giải pháp khả thi. Mỗi cá nhân được biểu diễn dưới dạng một chuỗi ký tự / số nguyên / float / bit. Chuỗi này tương tự như Chromosome.

+ Nền tảng thuật toán di truyền

Các thuật toán di truyền dựa trên sự tương tự với cấu trúc di truyền và hành vi của nhiễm sắc thể của quần thể. Sau đây là nền tảng của GAs dựa trên sự tương tự này:

1. Các cá thể trong quần thể cạnh tranh về tài nguyên và giao phối.
2. Những cá thể thành công (khỏe mạnh nhất) sau đó giao phối để tạo ra nhiều con cái hơn những người khác.
3. Các gen từ bố mẹ “phù hợp nhất” truyền trong suốt thế hệ, đôi khi bố mẹ tạo ra con cái tốt hơn cả bố và mẹ.
4. Do đó, mỗi thế hệ kế tiếp phù hợp hơn với môi trường của chúng.



+ Điểm thể lực Điểm thể lực được trao cho mỗi cá nhân, cho thấy khả năng “cạnh tranh” của một cá nhân . Cá nhân có điểm thể lực tối ưu (hoặc gần tối ưu) được tìm kiếm.

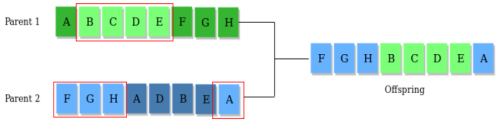
GAs duy trì quần thể n cá thể (nhiễm sắc thể / dung dịch) cùng với điểm thể lực của chúng. Những cá thể có điểm thể lực tốt hơn sẽ có nhiều cơ hội sinh sản hơn những cá thể khác. Những cá thể có điểm thể lực tốt hơn được chọn để giao phối và tạo ra con cái tốt hơn bằng cách kết hợp các nhiễm sắc thể của bố mẹ. Quy mô dân số là cố định nên phòng phải được tạo ra cho những người mới đến. Vì vậy, một số cá thể chết và bị thay thế bởi những người mới đến cuối cùng tạo ra thế hệ mới khi tất cả cơ hội giao phối của quần thể già bị cạn kiệt. Người ta hy vọng rằng qua các thế hệ kế tiếp, các giải pháp tốt hơn sẽ xuất hiện trong khi ít phù hợp nhất sẽ chết.

Mỗi thế hệ mới trung bình có nhiều “gen tốt hơn” so với cá thể (giải pháp) của các thế hệ trước. Do đó, mỗi thế hệ mới đều có "giải pháp từng phần" tốt hơn các thế hệ trước. Một khi con cái được tạo ra không có sự khác biệt đáng kể so với con cái được tạo ra bởi quần thể trước đó, thì quần thể đó sẽ được hội tụ. Thuật toán được cho là hội tụ vào một tập hợp các giải pháp cho vấn đề.

+ Toán tử của thuật toán di truyền

Khi thế hệ ban đầu được tạo ra, thuật toán sẽ phát triển thế hệ bằng cách sử dụng các toán tử sau:

1. Toán tử lựa chọn: Ý tưởng là ưu tiên những cá nhân có điểm thể lực tốt và cho phép họ truyền gen đó cho các thế hệ kế tiếp.
2. Toán tử Crossover: Điều này thể hiện sự giao phối giữa các cá thể. Hai cá nhân được chọn bằng cách sử dụng toán tử lựa chọn và các địa điểm giao nhau được chọn ngẫu nhiên. Sau đó, các gen tại các vị trí trao đổi chéo này được trao đổi để tạo ra một cá thể hoàn toàn mới (thế hệ con cái).



1. Toán tử đột biến: Ý tưởng chính là chèn các gen ngẫu nhiên vào đời con để duy trì sự đa dạng trong quần thể để tránh sự hội tụ sớm. 

Bài toán ví dụ và giải pháp sử dụng Thuật toán di truyền

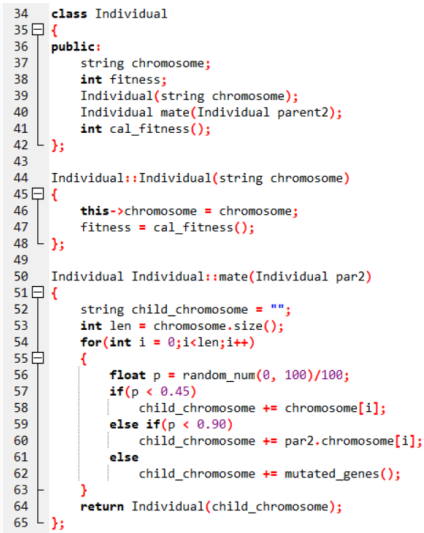
Cho một chuỗi mục tiêu, mục tiêu là tạo chuỗi mục tiêu bắt đầu từ một chuỗi ngẫu nhiên có cùng độ dài. Trong cách triển khai sau đây, các phép loại suy sau được thực hiện:

* Các ký tự AZ, az, 0-9 và các ký hiệu đặc biệt khác được coi là gen
* Một chuỗi được tạo bởi các ký tự này được coi là nhiễm sắc thể / giải pháp / Cá thể

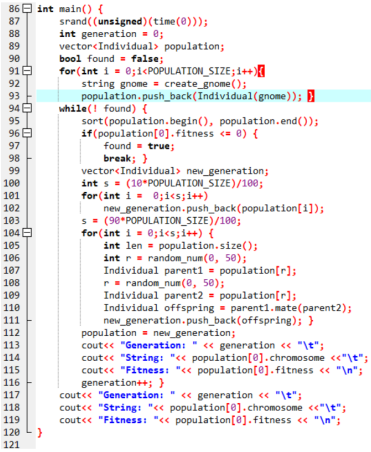
**Điểm thể chất** là số ký tự khác với các ký tự trong chuỗi mục tiêu ở một chỉ mục cụ thể. Vì vậy, cá nhân có giá trị thể lực thấp hơn được ưu tiên hơn.

1. Cài đặt chương trình







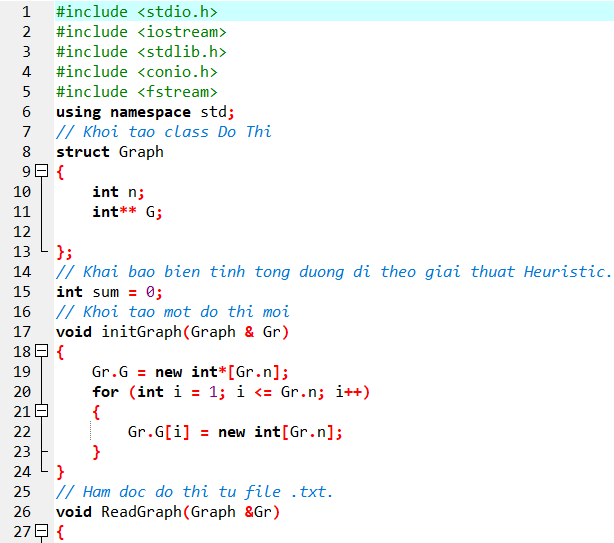


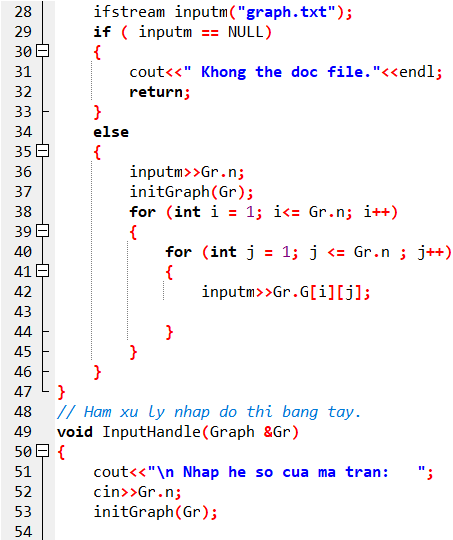
1. Cài đặt thuật toán Heuristic

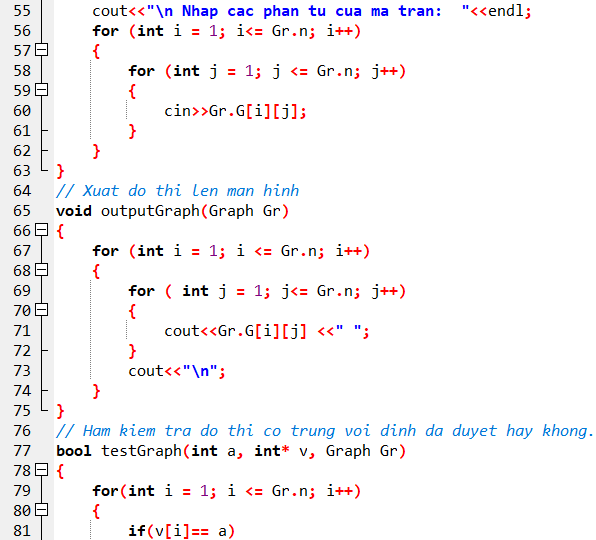
1. Cơ sở lý thuyết

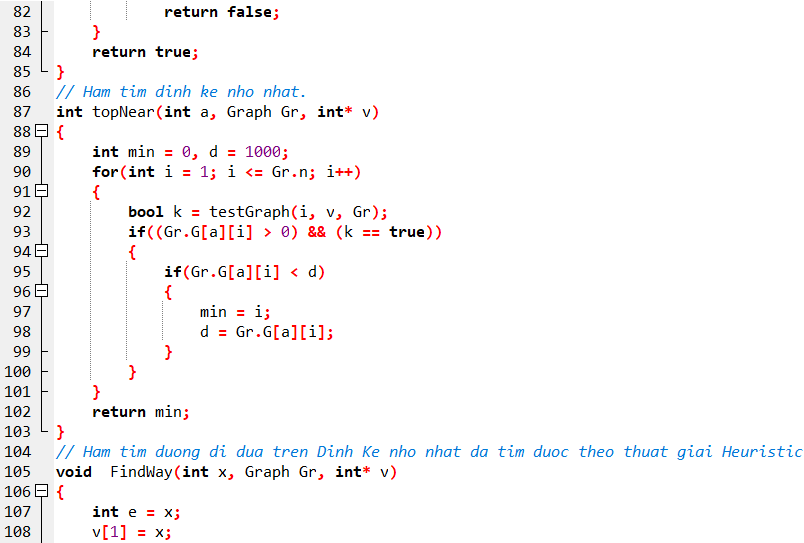
* Là các kỹ thuật dựa trên kinh nghiệm để giải quyết vấn đề, học hỏi hay khám phá nhằm đưa ra một giải pháp mà không được đảm bảo là tối ưu. Với việc nghiên cứu khảo sát không có tính thực tế, các phương pháp heuristic được dùng nhằm tăng nhanh quá trình tìm kiếm với các giải pháp hợp lý thông qua các suy nghĩ rút gọn để giảm bớt việc nhận thức vấn đề khi đưa ra quyết định.

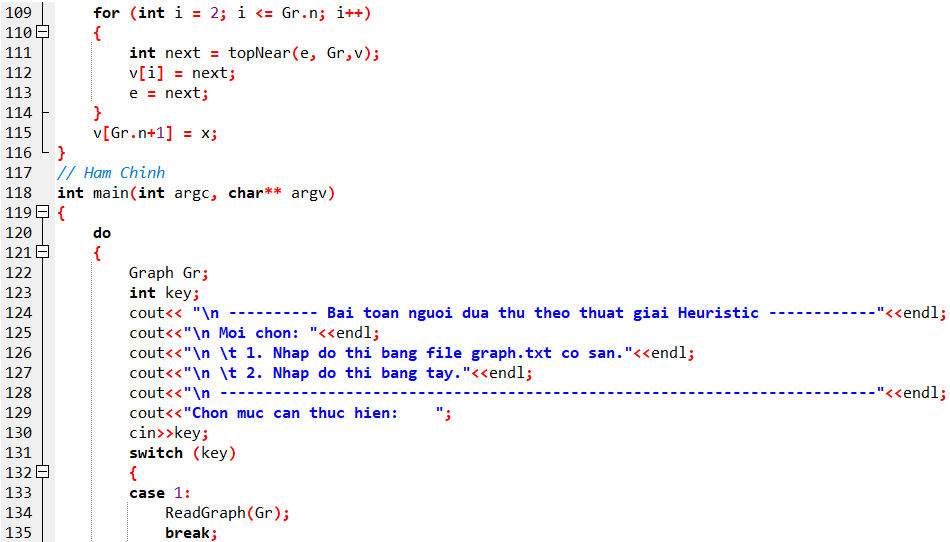
2. Cài đặt chương trình







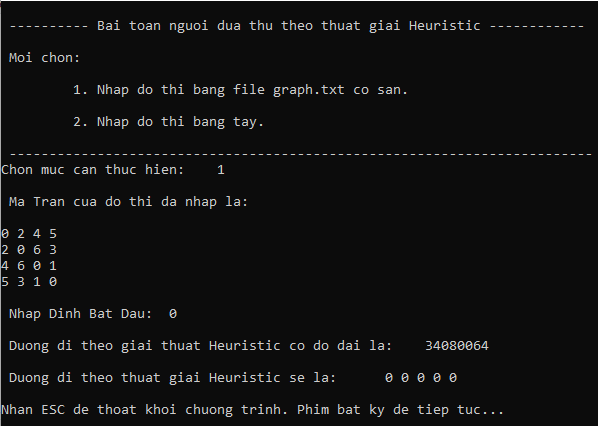








3. Kết quả



LINK SOURCE CODE

(Truy cập bằng mail trường để lấy vào link)

1. Bài toàn Big Number

<https://drive.google.com/file/d/1fELRnyJhAMvpJv6XRrO8CiCCr0SnHjDj/view?usp=sharing>

1. Thuật toán Suy dẫn tiến

<https://drive.google.com/file/d/1YxyodLZ_ayIOm8Z8hPikXbOpb1itxX7b/view?usp=sharing>

1. Game Bóc sỏi

<https://drive.google.com/file/d/1S3oowGEquwYe9FN72zuHSC9MsBsjLzMS/view?usp=sharing>

1. Game Tic-tac-toe

<https://drive.google.com/file/d/150o5zr2LoRejy0kiZf1KWprxQlMfT5Sv/view?usp=sharing>

1. IoT - Đồng hồ thời gian thực

<https://drive.google.com/drive/folders/1RpmHIxcV3Yv2aelmrD0h6phqTiei-fap?usp=sharing>

1. Bài toán N-queen

<https://drive.google.com/drive/folders/1d1AWejYepSm8GLhZtYIcK2RTg1-INKDx?usp=sharing>

1. Thuật toán Robinson

<https://drive.google.com/drive/folders/18YmPIlxa0tdf566g-vt4txq4v2aGvNFf?usp=sharing>

1. Thuật toán Vương hạo

<https://drive.google.com/file/d/1hjoA_js3ve4W7I_1ZTPRaWG8YMl9AuV_/view?usp=sharing>

1. Bài toán tam giác

<https://drive.google.com/file/d/1fjfpEfzOFyMdPSt2E7DCLdKD8Nbo6-9p/view?usp=sharing>

1. Thuật toán di truyền

<https://drive.google.com/file/d/1RahafcUDbOFPae4twcp9UDSjWJ7uvMNA/view?usp=sharing>

1. Thuật toán Heuristic

<https://drive.google.com/drive/folders/1CUXsbJHIZ8ezNkMMoF43Y2DpydrmasPP?usp=sharing>