

Asymptote tutorial for beginners

Trần Quân ++ ...

Bài viết này sẽ hướng dẫn việc cài đặt, dịch và lập trình cơ bản **Asymptote** cho người mới bắt đầu.

Mục lục

1 Cài đặt Asymptote và các chương trình	2
1.1 Asymptote trên Windows	2
1.1.1 Cài đặt	2
1.1.2 Dịch ASY ở cửa sổ lệnh	4
1.1.3 ASY với Notepad++	4
1.1.4 ASY với TexStudio	6
1.2 Asymptote trên Ubuntu	9
1.3 Asymptote trên MacOS	9
2 Vẽ hình học phẳng với ASY cơ bản	11
2.1 Kiểu dữ liệu	12
2.2 Các phép biến hình	12
2.3 Điểm	13
2.4 Đoạn thẳng, đường thẳng	14
2.5 Đường tròn, cung tròn, E-líp	14
2.6 Giao điểm	15
2.7 Một số ví dụ	17
3 Tham khảo	26

Asymptote (vector graphics language) là ngôn ngữ lập trình đồ họa dạng vector được thiết kế để tạo hình vẽ và biểu đồ toán học. Asymptote có thể xuất hình ảnh ở định dạng ép hoặc pdf và tương thích với Latex. Asymptote được lấy cảm hứng từ MetaPost, tuy nhiên nó là một ngôn ngữ lập trình hoàn chỉnh và cú pháp rõ ràng hơn (giống ngôn ngữ C++) so với Metapost.

Asymptote có thể chạy trên các hệ điều hành thông dụng như Windows, Linux và MacOS.

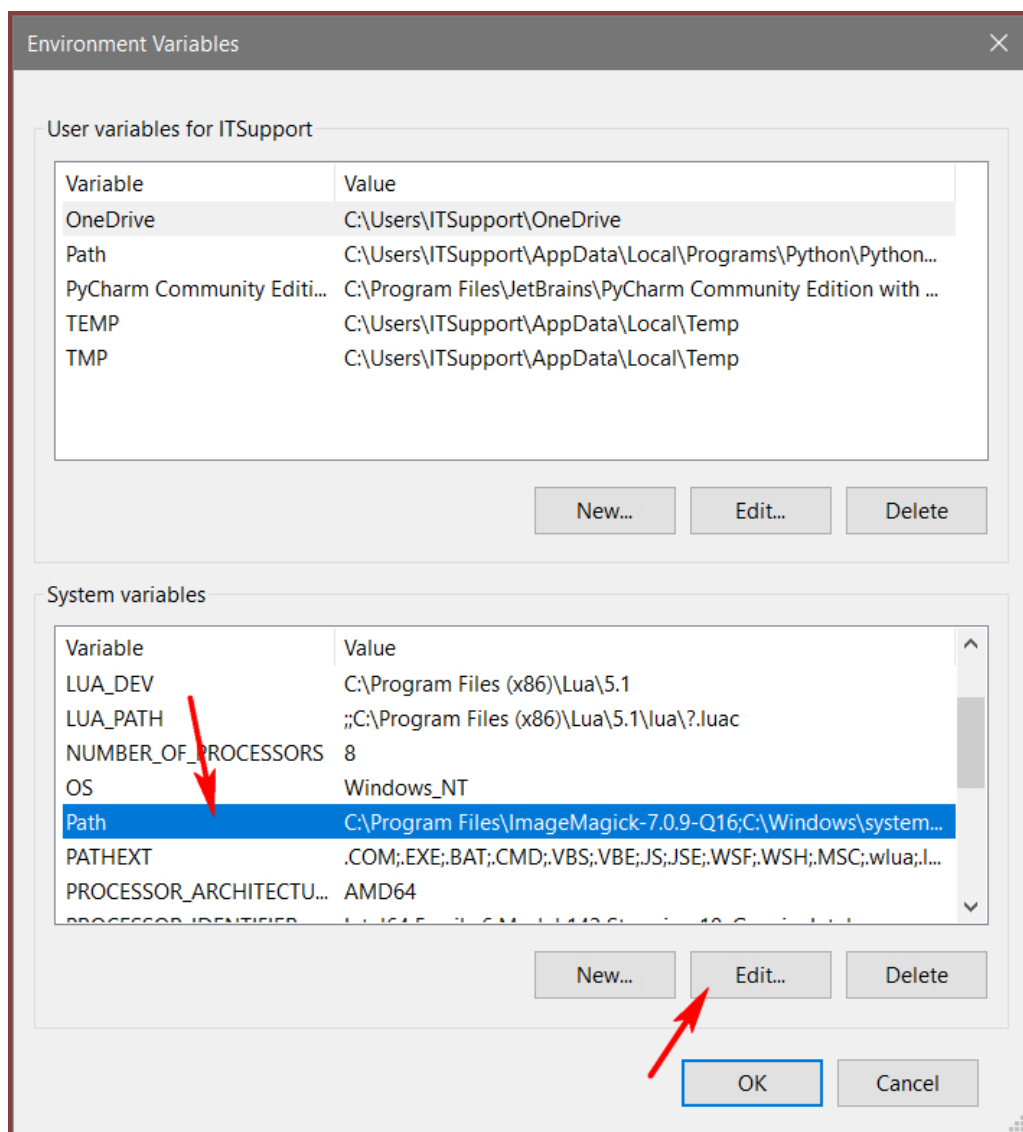
1 Cài đặt Asymptote và các chương trình

1.1 Asymptote trên Windows

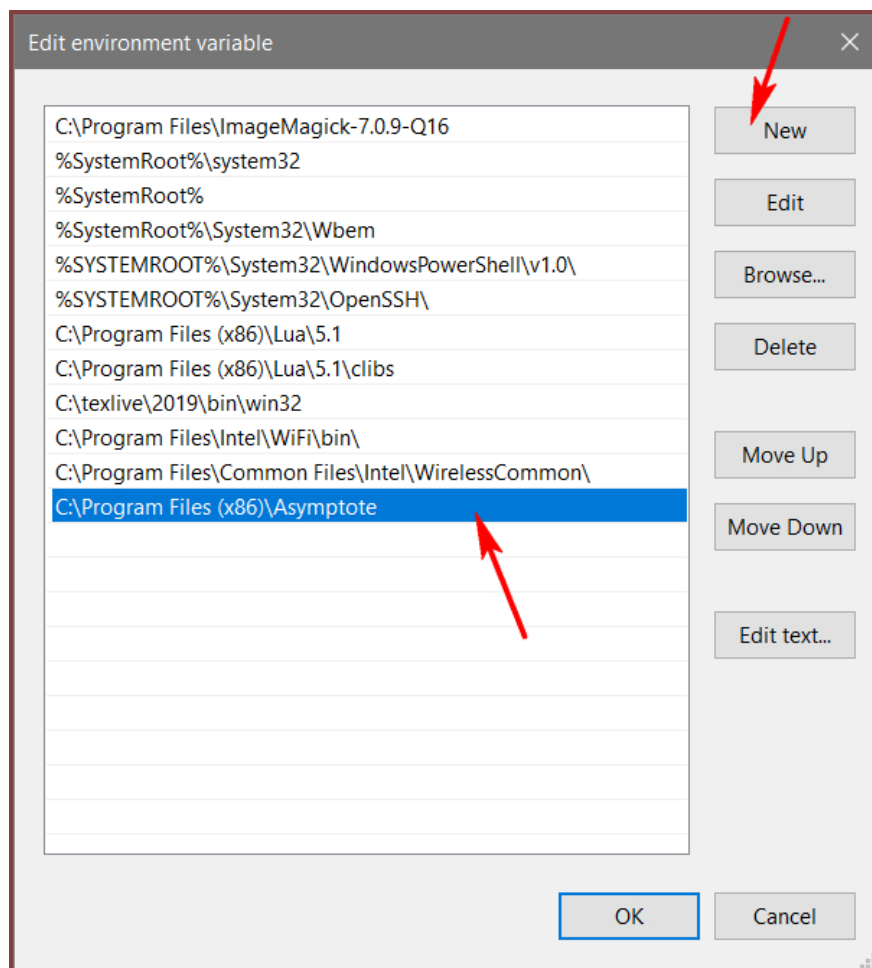
1.1.1 Cài đặt

Download phiên bản mới nhất của **Asymptote** theo link <http://sourceforge.net/projects/asymptote> và tiến hành cài đặt. Sau khi cài, thư mục cài trên Windows là C:\ProgramFiles(x86)\Asymptote.

Sau khi cài đặt, thêm đường dẫn trên vào biến **PATH** của Windows như sau: *Control Panel* → *System* → *Advanced system settings*. Sau đó chọn tab **Advanced** và chọn **Environment Variables**.



Trên hộp thoại **Environnement Variables**, chọn **Path**, sau đó chọn **Edit**. Xuất hiện hộp thoại **Edit environment variables**.



Trên hộp thoại **Edit environment variables**, chọn **New** để thêm đường dẫn `C:\ProgramFiles(x86)\Asymptote`.

Sau đó chạy ra cửa sổ lệnh của *Windows*, gõ lệnh **PATH** để xác nhận đường dẫn này. Khi đó gõ lệnh *asy* (chạy file *asy.exe*), trên cửa sổ lệnh sẽ như sau:

```

Administrator: Command Prompt - asy
11/18/2019 12:27 PM          23,846 three_arrows.asy
11/18/2019 12:27 PM           3,673 three_light.asy
11/18/2019 12:27 PM           2,764 three_margins.asy
11/18/2019 12:27 PM          73,252 three_surface.asy
11/18/2019 12:27 PM          12,522 three_tube.asy
11/18/2019 12:27 PM           1,375 tree.asy
11/18/2019 12:27 PM           5,797 trembling.asy
11/18/2019 12:27 PM           4,297 tube.asy
11/18/2019 12:27 PM             51 unicode.asy
11/19/2019 07:56 AM          51,652 uninst.exe
11/18/2019 12:41 PM           23 version.asy
11/12/2019 04:17 PM    <DIR>      webgl
11/18/2019 12:27 PM          4,812 x11colors.asy
11/18/2019 12:41 PM           90 xasy
    108 File(s)      14,385,844 bytes
     6 Dir(s)  244,788,703,232 bytes free

C:\Program Files (x86)\Asymptote>asy
Welcome to Asymptote version 2.61 (to view the manual, type help)
>

```

Đây chính là **interactive mode** của ASY. Lưu ý, để thoát khỏi **interactive mode**, có thể gõ lệnh **quit** hoặc dùng tổ hợp phím **Ctrl + C**.

1.1.2 Dịch ASY ở cửa sổ lệnh

Dùng một chương trình text editor bất kỳ để tạo file **hd01.asy** có nội dung như sau:

```
unitsize(1cm);

pair A=(1.5,2.5); dot(Label("$A$"),align=NE,A);
pair B=(0,0); dot(Label("$B$"),align=SE,B);
pair C=(7,0); dot(Label("$C$"),align=SE,C);

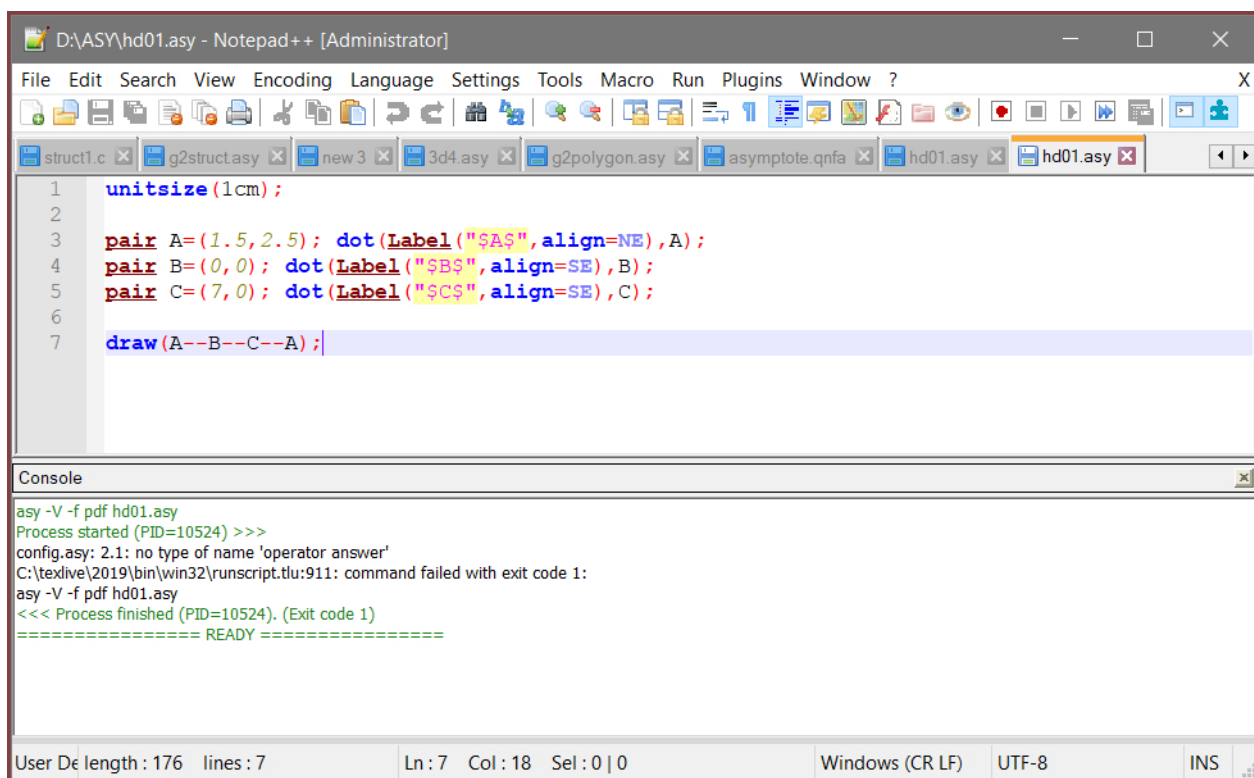
draw(A--B--C--A);
```

Ở cửa sổ lệnh của *Windows*, gõ lệnh: **asy -V -f pdf hd01.asy**. Khi đó chương trình **asy.exe** sẽ chạy và tạo ra file **hd01.pdf** cùng thư mục với file **hd01.asy** và gọi phần mềm đọc file pdf để mở nó.

1.1.3 ASY với Notepad++

Để thuận tiện cho việc lập trình ASY, có thể dùng *text editor* rất quen thuộc với những người lập trình là **Notepad++**.

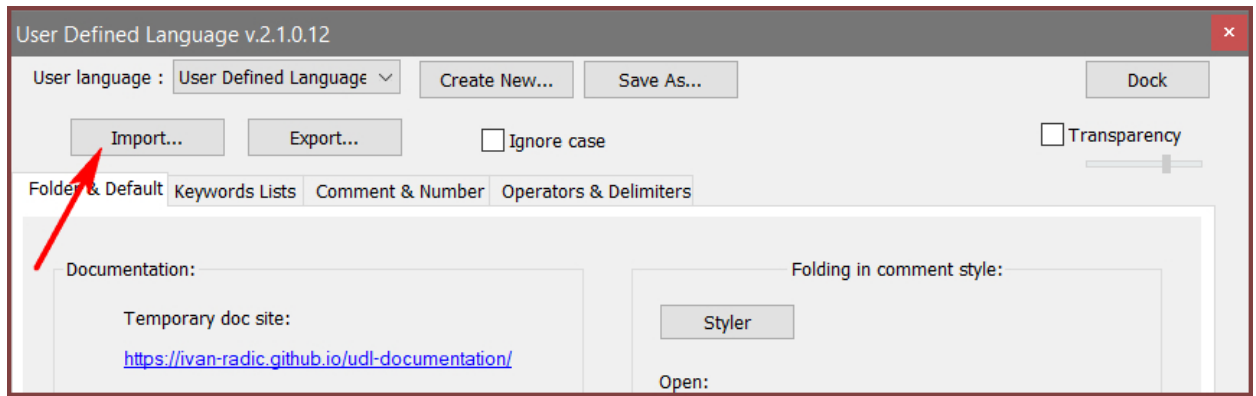
Download **Notepad++** theo link <https://notepad-plus-plus.org/downloads/> và cài đặt nó. Giao diện chương trình như sau:



Tiếp theo, chúng ta cần *highlight* cú pháp của *ASY* trên *Notepad++*.

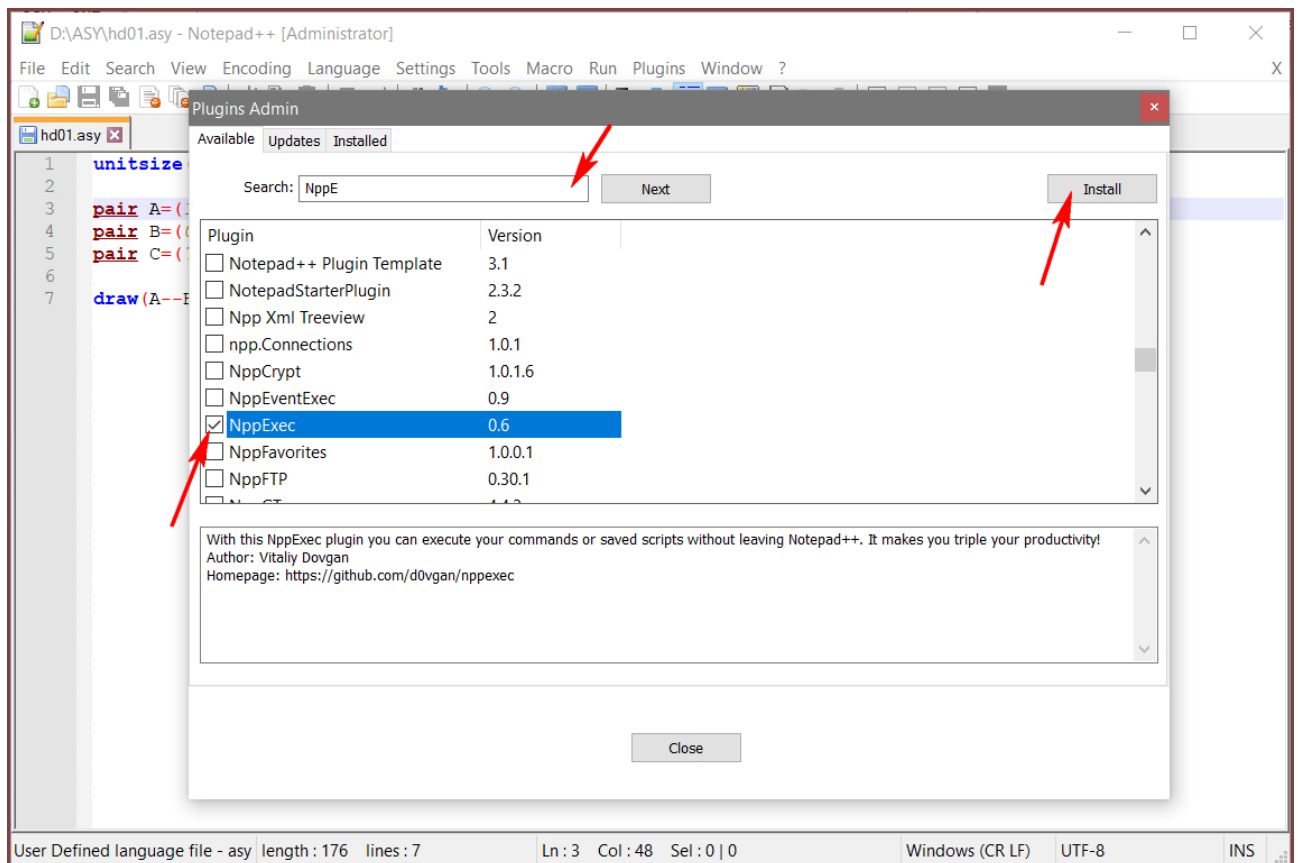
Download file **UserDefineLang.xml** theo link <https://github.com/asymarris/asy-coloration-dans-npp>.

Trên **Notepad++**, chọn menu *Language* → *Define you language*, xuất hiện hộp thoại **User Defined Language**.



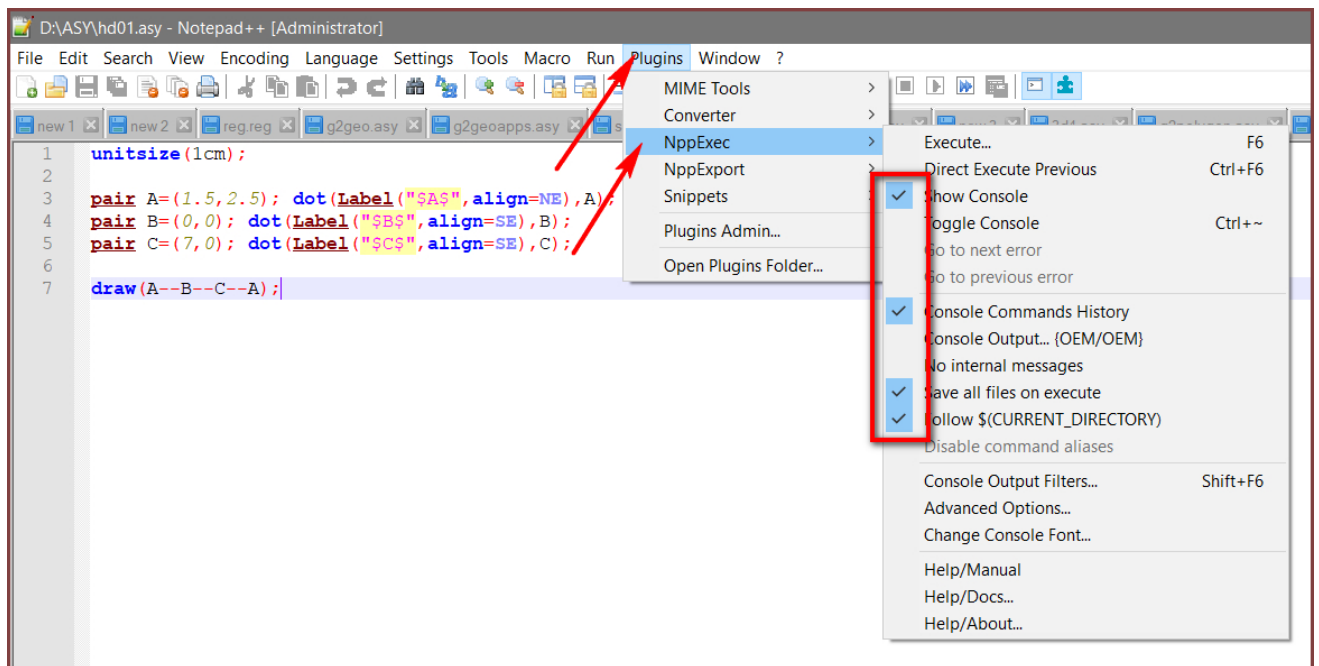
Chọn **Import**, sau đó chọn file **UserDefineLang.xml**. Sau khi kết thúc, kiểm tra lại việc *highlight* các từ khóa của **ASY**.

Notepad++ cho phép chạy các chương trình bên ngoài thông qua plugin **NppExec**. Để cài đặt plugin này, chọn menu **Plugins** → **Plugins Admin**.

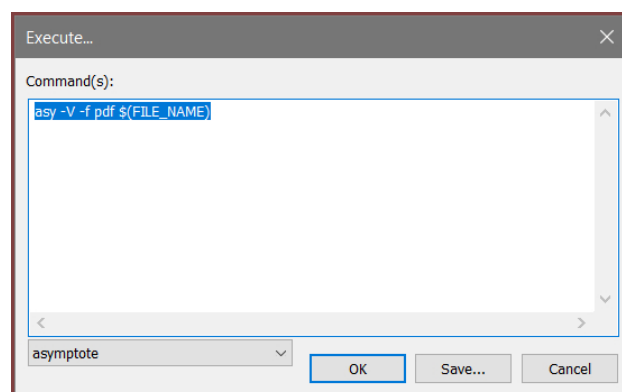


Gõ **NppExec** vào ô **Search**, sau đó nhấp nút **Next** và check **NppExec** vào ô chọn. Sau đó nhấp nút **Install** để cài đặt plugin này. **Notepad++** sẽ cài đặt **NppExec** và chạy lại.

Khi đó trên menu, chọn **Plugins**, sẽ xuất hiện thêm lựa chọn **NppExec**. Lưu ý chọn vào các lựa chọn: *Show Console*, *Console Commands History*, *Save all files on execute*, *Follow*.



Chọn **Excute ... (F6)**.



Nhập lệnh sau vào ô **Command(s)**, sau đó nhấp nút **Save** để lưu lại. Bấm **OK** để thực hiện lệnh này.

```
asy -V -f pdf $(FILE_NAME)
```

Trên cửa sổ **Console** của **Notepad++** sẽ xuất hiện quá trình thực thi lệnh và sẽ gọi chương trình đọc file pdf (ví dụ Acrobat Reader) trên *Windows* để mở file sau khi tạo.

1.1.4 ASY với TexStudio

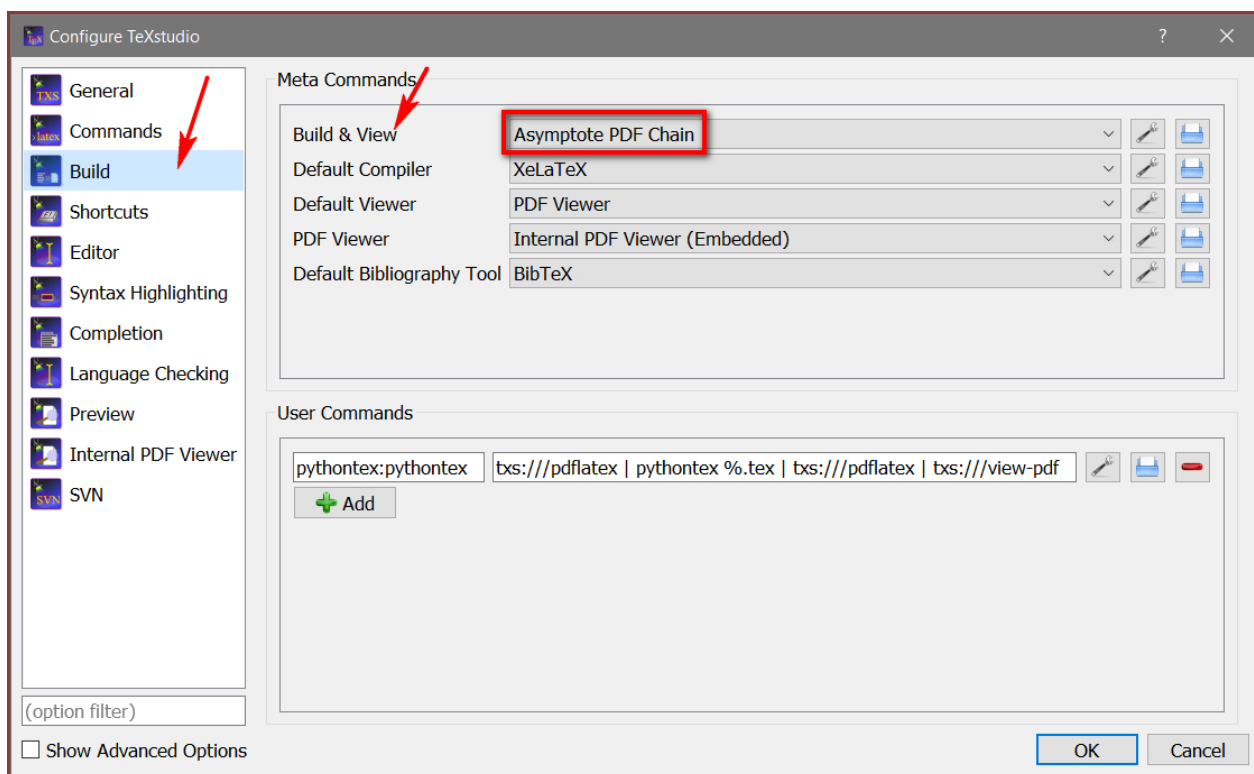
TexStudio là trình soạn thảo latex có thể chạy trên các hệ điều hành thông dụng như Windows, Linux, MacOS. TexStudio được cài đặt cùng với **TexLive** và **Ghostscript**. Download và cài đặt theo link:

- TexStudio: <https://www.texstudio.org/>.
- TexLive: Nên download file iso để việc cài đặt thuận tiện hơn: <https://www.tug.org/texlive/acquire-iso.html>.
- Ghostscript: <https://www.ghostscript.com/download/gsdnld.html>.

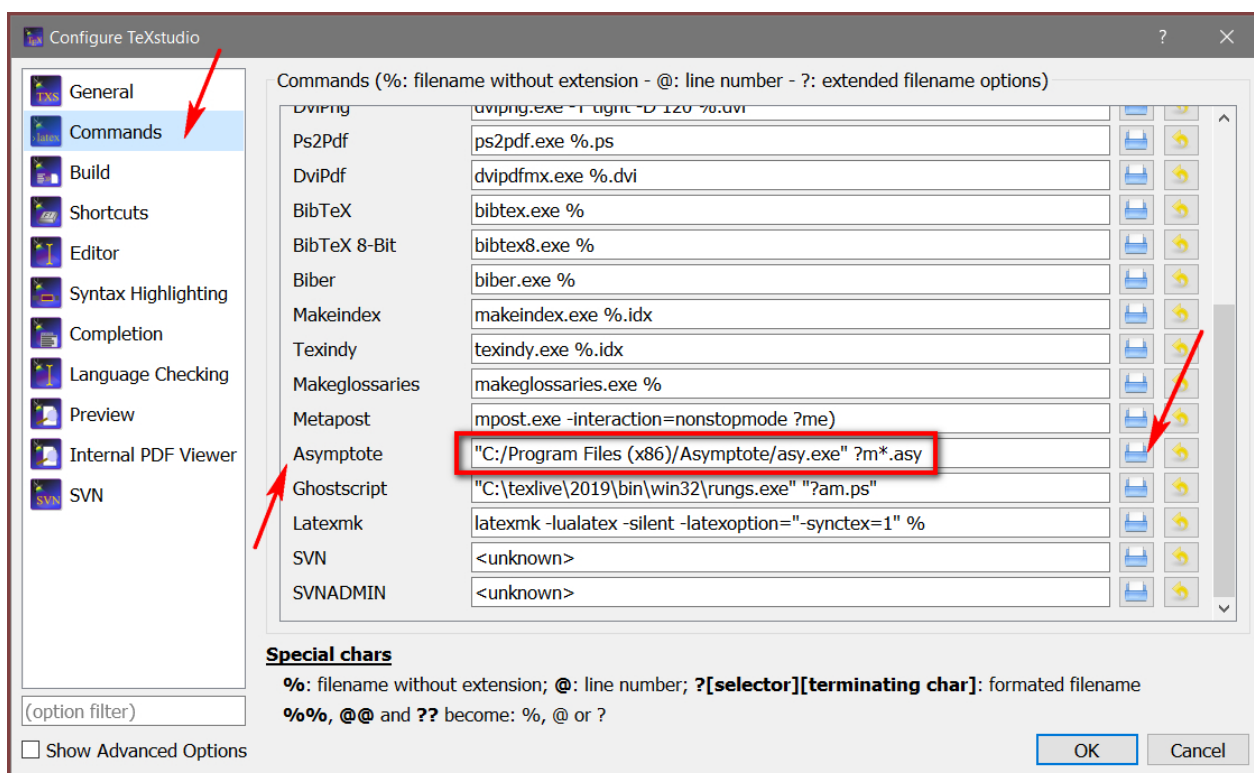
Ngoài ra nên cài thêm **ImageMagick** khi cần chuyển đổi file pdf thành file ảnh.

Lưu ý, **TexLive** đã có sẵn gói **ASY**. Tuy nhiên phiên bản **ASY** trên **TexLive** cập nhật chậm. Nên download **ASY** và cài đặt theo hướng dẫn ở mục 1, việc cấu hình để sử dụng **ASY** sẽ được hướng dẫn tiếp theo.

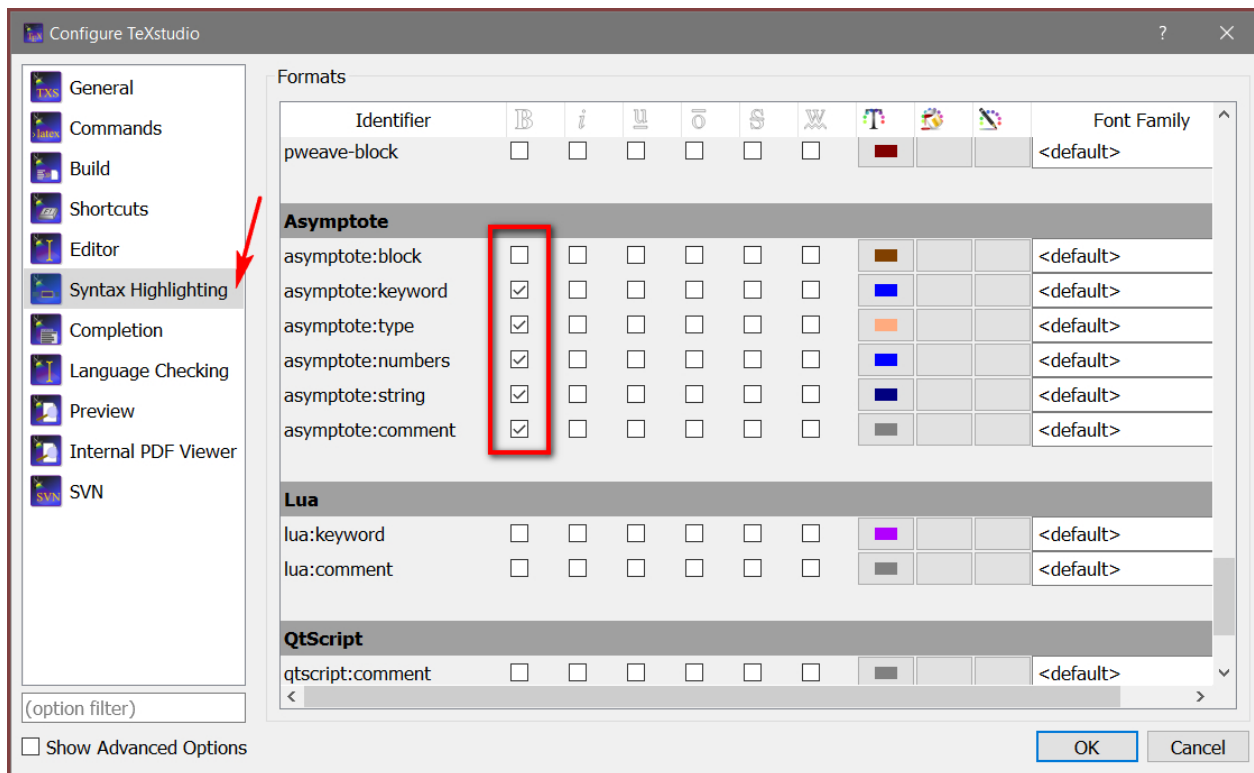
Để dịch **ASY** trên **TexStudio**, trên giao diện **TexStudio** chọn menu *Options* → *Configure TexStudio*. Sau đó chọn **Build** và ở lựa chọn **Build View**, chọn **Asymptote PDF Chain**.



Như đã nói trên, **TexLive** có sẵn **asy.exe** ở thư mục **C:\texlive\2019\bin\win32** và thư mục này được khai báo trên Windows PATH. Để chọn file **asy.exe** khác, chọn **Commands**, ở lựa chọn **Asymptote**, chọn đường dẫn đến file **asy.exe** theo đường dẫn **C:\Program Files (x86)\Asymptote**.



Sau bước này chúng ta đã có thể dịch các chương trình **ASY** trên **TexStudio**. Tuy nhiên để việc viết mã **ASY** được thuận tiện, cần cấu hình **Syntax Highlight** cho các từ khóa của **ASY**. Chọn **Syntax Highlighting**, ở mục **Asymptote**, check vào các ô cần thiết như *keyword*, *type*, *number*, *string*, *comment*.



Tuy nhiên ngầm định **Syntax Highlight** của **TeXStudio** cho **ASY** khá ít từ khóa được highlight, cần bổ sung các thêm các từ khóa.

Download file **asymptote.qnfa** theo link <https://github.com/texstudio-org/texstudio/blob/master/utilities/qxs/asymptote.qnfa>. Sau đó tạo thư mục **config\languages** trong thư mục **C:\ProgramFiles(x86)\texstudio** và copy file **asymptote.qnfa** vào thư mục đó. Đường dẫn sau khi copy **C:\ProgramFiles(x86)\texstudio\config\languages\asymptote.qnfa**.

Nếu sử dụng **TeXStudio** bản **portable**, sử dụng đường dẫn sau **\texstudio-2.12.16-portable\config\languages\asymptote.qnfa**.

File **asymptote.qnfa** là file text có cấu trúc. Để thêm các từ khóa, mở nó bằng bất cứ chương trình text editor nào và thêm các từ khóa cần thiết.

Cấu hình cho **TeXStudio** sử dụng được **ASY** nói trên thực chất **TeXStudio** tạo file **.asy** và gọi **asy.exe** để dịch nó thành file **pdf**, sau đó nhúng vào file **pdf** chính của tex.

Ví dụ, file **QHT122019.tex** có chứa mã lệnh **ASY**. Khi **TeXStudio** dịch file này, nó sẽ tạo mỗi mục **ASY** thành các file **QHT122019-1.asy**, **QHT122019-2.asy**, ... và gọi **asy.exe** để dịch các file này thành file pdf. Sau đó nhúng vào file **QHT122019.pdf** chính.

Cách thực hiện này có thể gặp lỗi trong các trường hợp sau:

- File pdf đang mở ở chương trình khác nên **asy.exe** không tạo được file pdf đè lên.
- Trong file tex có nhiều mục **ASY**, khi thêm, xóa các file này, có thể **TeXStudio** không đè lên file cũ. Trường hợp này, có thể xóa các file không cần thiết và chỉ cần giữ lại file tex, sau đó dịch lại.

Lưu ý, một mục **ASY** trên **TeXStudio** nằm trong cặp **begin**, **end** như sau:

```
\begin{asy}

unitsize(1cm);
pair A=(1.5,2.5); dot(Label("$A$"),align=NE,A);
pair B=(0,0); dot(Label("$B$"),align=SE,B);
pair C=(7,0); dot(Label("$C$"),align=SE,C);
draw(A--B--C--A);

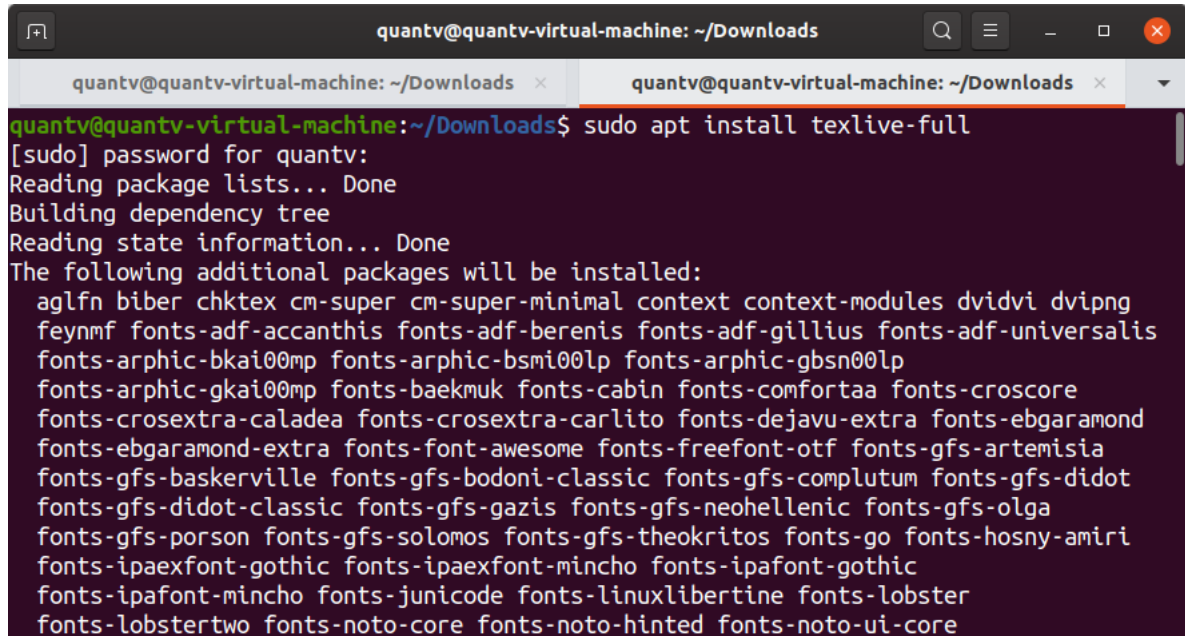
\end{asy}
```


1.2 Asymptote trên Ubuntu

Trên **Ubuntu**, nên theo các bước sau: cài TexLive, cập nhật ASY và cài TexStudio và cấu hình.

Để cài đặt **TexLive**, chạy **Terminal** của **Ubuntu** và gõ lệnh sau:

```
$ sudo apt install texlive-full
```



```
quantv@quantv-virtual-machine: ~/Downloads
quantv@quantv-virtual-machine: ~/Downloads
quantv@quantv-virtual-machine:~/Downloads$ sudo apt install texlive-full
[sudo] password for quantv:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
 aglfn biber chktex cm-super cm-super-minimal context context-modules dvidvi dvipng
 feynmf fonts-adf-accanthis fonts-adf-berenis fonts-adf-gillius fonts-adf-universalis
 fonts-arphic-bkai00mp fonts-arphic-bsmi00lp fonts-arphic-gbsn00lp
 fonts-arphic-gkai00mp fonts-baekmuk fonts-cabin fonts-comfortaa fonts-croscore
 fonts-crosextra-caladea fonts-crosextra-carlito fonts-dejavu-extra fonts-ebgaramond
 fonts-ebgaramond-extra fonts-font-awesome fonts-freefont-otf fonts-gfs-artemisla
 fonts-gfs-baskerville fonts-gfs-bodoni-classic fonts-gfs-complutum fonts-gfs-didot
 fonts-gfs-didot-classic fonts-gfs-gazis fonts-gfs-neohellenic fonts-gfs-olga
 fonts-gfs-porson fonts-gfs-solomos fonts-gfs-theokritos fonts-go fonts-hosny-amiri
 fonts-ipaexfont-gothic fonts-ipaexfont-mincho fonts-ipafont-gothic
 fonts-ipafont-mincho fonts-junicode fonts-linuxlibertine fonts-lobster
 fonts-lobstertwo fonts-noto-core fonts-noto-hinted fonts-noto-ui-core
```

Tương tự như Windows, TexLive đã có ASY. Trong trường hợp bản ASY trên TexLive chưa cập nhật, có thể dùng lệnh **tlmgr update --self** để cập nhật cho TexLive hoặc download bản **deb** mới nhất của ASY theo link sau: <https://launchpad.net/ubuntu/+source/asymptote>.

ASY mới nhất hiện tại là **asymptote_2.61-1_amd64.deb**. Khi download về thường trong thư mục **Downloads**. Chạy **Terminal** của Ubuntu và gõ lệnh sau:

```
$ cd Downloads
$ sudo dpkg -i asymptote_2.61-1_amd64.deb
```

Lệnh này sẽ cài và cập nhật ASY bản mới nhất.

Để cài TexStudio, ở **Terminal** gõ lệnh:

```
$ sudo apt install texstudio
```

Việc cấu hình **Asymptote** trên **TexStudio** tương tự như trên Windows.

1.3 Asymptote trên MacOS

Cài **ASY** trên **MacOS** có nhiều cách: Cài (dịch) từ *source*; Cài qua gói **MacTex** (phiên bản trên **MacOS** của **TexLive**) và Cài thông qua **MacPorts**.

Để dịch từ *source*, download file *source* theo link <https://asymptote.sourceforge.io/>. Ta được file **asymptote-x.x.x.src.tar** (x.x.x là phiên bản, ví dụ **asymptote-2.6.1.src.tar**). File này được lưu vào thư mục **Downloads**.

Chạy **Terminal** của **MacOS** và chạy các lệnh sau:

```
$ cd Downloads
$ tar -xzf asymptote-2.6.1.src.tar
$ cd asymptote-2.6.1
```

```
$ sudo make all
$ sudo make install
```

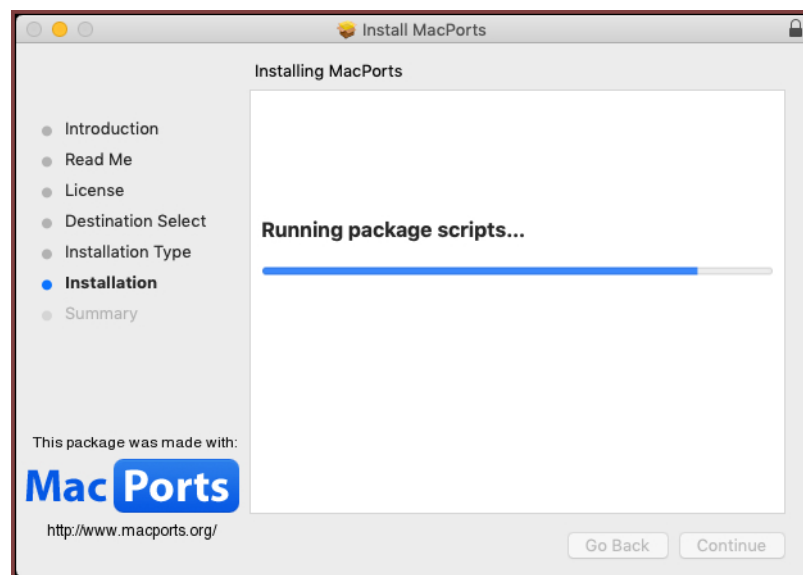
Lưu ý khi chạy `./configure` có thể được yêu cầu cài đặt các trình biên dịch **Xcode** hoặc **gcc**, chọn cài **gcc**.

Tuy nhiên dịch từ *source* dễ gặp lỗi vì thiếu **package** và khó xử lý.

Để cài đặt **MacTeX**, download file **MacTeX.pkg** theo link <http://www.tug.org/mactex/downloading.html>. Kết thúc download, nhấp đúp vào file này để cài đặt.

Sau khi cài đặt **MacTeX**, kiểm tra phiên bản ASY bằng cách chạy lệnh **asy** ở **Terminal**. Trong trường hợp ASY chưa được cập nhật mới, vào Launchpad, chạy chương trình **TeXLive Utility** để cập nhật các gói mới.

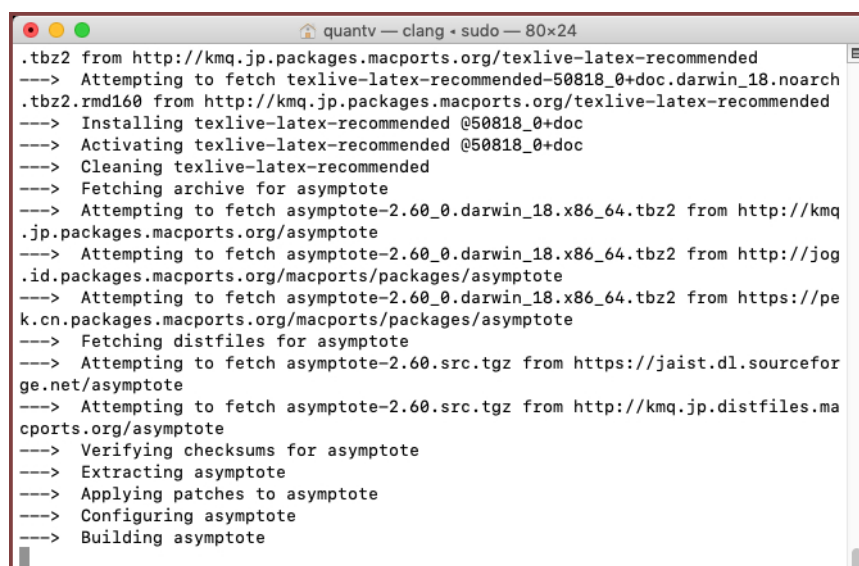
ASY cũng có thể được cài đặt qua **MacPorts**. Download **MacPorts** cho phiên bản **MacOS** tương ứng theo link <https://www.macports.org/install.php>. Trong ví dụ này là bản **macOS Mojave v10.14**. File download được là **MacPorts-2.6.2-10.14-Mojave.pkg** lưu vào thư mục **Downloads**. Sau khi *download*, nhấp đúp vào file này để tiến hành cài đặt MacPorts. Trong quá trình cài đặt có thể yêu cầu cài **XCode** hoặc **gcc**.



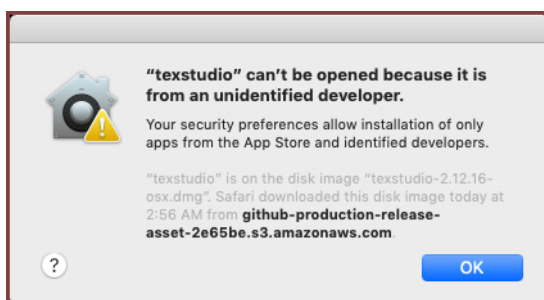
Kết thúc cài **MacPorts**, chạy **Terminal** của MacOS và chạy lệnh sau để cài đặt **asymptote**:

```
$ sudo port install asymptote
```

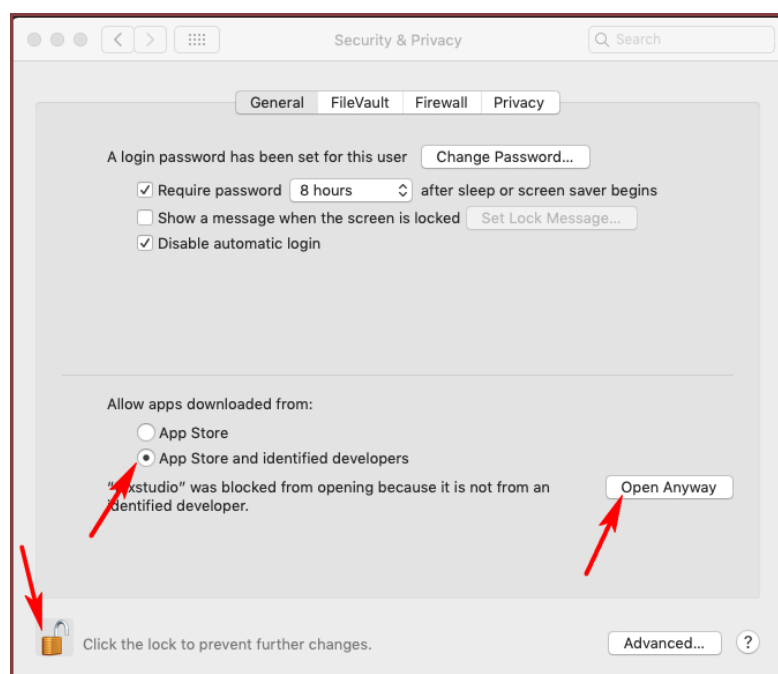
Khi đó **MacPorts** sẽ download source **asymptote-x.x.x.src.tar** và khá nhiều gói cần thiết trong đó có cả **texlive** nên không cần cài **TeXLive**. Kết thúc cài đặt, chạy lệnh **asy** để kiểm tra phiên bản *asy* nào.



Download **TexStudio** theo link <https://www.texstudio.org/#download>. Sau khi download ta được file **texstudio-x.x.x-osx.dmg** (x.x.x là phiên bản của **TexStudio**, ví dụ **texstudio-2.12.16-osx.dmg**). Nhấp đúp vào file để mở file *dmg*, sẽ xuất hiện chương trình **TexStudio**, nhấp đúp để chạy nó.



Nếu có thông báo **unidentified developers** (ứng dụng của nhà phát triển chưa được xác thực), chọn menu *System Settings* → *Security & Privacy*. Ở tab **General** nhấp vào biểu tượng khóa để mở khóa. Sau đó chọn **App Store and identified developers**.



Việc cấu hình **Asymptote** trên **TexStudio** tương tự như trên **Windows**, tuy nhiên lưu ý chọn menu là *TexStudio* → *Preferences*.

Trên MacOS, file **asymptote.qnfa** được lưu trong thư mục `~/.config/texstudio/languages/asymptote.qnfa`.

2 Vẽ hình học phẳng với ASY cơ bản

ASY có gói **geometry.asy** dành riêng cho vẽ hình học. Trước khi hướng dẫn sử dụng gói **geometry.asy**, chúng tôi hướng dẫn vẽ hình học phẳng các lệnh cơ bản của ASY.

Để vẽ hình học phẳng, cần chú ý những vấn đề sau:

- Điểm, đoạn thẳng, đường thẳng, đường tròn, cung tròn.
- Đường vuông góc, đường song song, đường trung trực, đường phân giác.
- Các phép biến hình như tịnh tiến, vị tự, quay, đối xứng trục.

Để thuận tiện cho việc vẽ hình, chúng tôi cung cấp gói **geo2d.asy** bao gồm các hàm được lập trình dựa trên các lệnh cơ bản của ASY.

2.1 Kiểu dữ liệu

Khi vẽ hình học cần quan tâm đến các kiểu dữ liệu sau:

a. void

Kiểu dữ liệu void chỉ được sử dụng bởi các hàm mà không trả về bất cứ giá trị nào.

b. bool

Kiểu boolean chỉ có giá trị true hoặc false.

c. int

Kiểu nguyên là một số nguyên nằm trong khoảng intMin và inMax. Nếu không chọn giá trị nào thì sẽ sử dụng số nguyên 0.

d. real

Kiểu thực là một số thực nằm trong khoảng realMin và realMax có độ chính xác realEpsilon. Nếu không chọn giá trị nào thì sẽ sử dụng số thực 0.0.

e. pair

Kiểu điểm được xác định bằng tọa độ (x,y).

f. string

Kiểu chuỗi là một chuỗi ký tự nằm trong cặp (").

g. path

Trên ASY, một path (đường) là tập hợp của các điểm (pair) hoặc các đường (path) được nối với nhau bằng – (đoạn thẳng) hoặc .. (đường con Bezier). Khi vẽ hình học phẳng chủ yếu dùng –.

Các hàm circle, arc, ellipse, ... đều sử dụng kiểu dữ liệu path.

2.2 Các phép biến hình

a. shift - phép di chuyển

```
transform shift(pair z);      Di chuyển đến điểm z
transform shift(real x, real y); Di chuyển đến điểm có tọa độ (x,y)
```

Ví dụ:

```
path c0=shift((2,4))*unitcircle;      Di chuyển đường tròn đơn vị đến điểm có tọa độ (2,4).
```

b. scale (xscale, yscale) - phép biến hình tỷ lệ

```
transform xscale(real x);      Biến hình theo trục x theo tỷ lệ x.
transform yscale(real y);      Biến hình theo trục y theo tỷ lệ y.
transform scale(real s);       Biến hình theo cả 2 trục với tỷ lệ s.
transform scale(real x, real y); Biến hình theo trục x tỷ lệ x, trục y tỷ lệ y.
```

c. rotate - phép quay

`transform rotate(real angle, pair z=(0,0));` Phép quay tâm z, góc angle.

d. reflect - phép đối xứng qua đường thẳng

`transform reflect(pair a, pair b);` Phép đối xứng qua đường thẳng ab.

e. scale - phép vị tự (geo2d.asy)

`transform scale(pair c, real k);` Phép vị tự tâm c, tỷ lệ k.

Cách sử dụng các phép biến hình này sẽ được chi tiết trong các ví dụ tiếp theo.

2.3 Điểm

Điểm (pair) trên ASY được khai báo như sau:

`pair A=(x,y);` Khai báo điểm A có tọa độ (x,y).

`pair A[];` Khai báo mảng các điểm A[i], bắt đầu từ A[0]. Sau đó có thể gán giá trị cho các điểm.

Có thể định nghĩa điểm theo một số cách sau:

`pair midpoint(path p);` Trả về điểm chính giữa của path p.

`pair M=midpoint(A--B);` Trả về điểm M là trung điểm của đoạn AB.

`pair M=midpoint(arc(O,B,C));` Trả về điểm M là điểm chính giữa cung \widehat{BC} , tâm O.

`pair M=(A+0.3*abs(A-B));` Trả về điểm M trên AB sao cho $AM = 0.3AB$. Hàm *abs(A-B)* trả về độ dài đoạn thẳng AB.

`pair Q=projection(P,A,B);` Trả về điểm Q là hình chiếu vuông góc của điểm P lên A-B (gói geo2d.asy).

`pair D=reflect(B,C)*A;` Trả về điểm D đối xứng với A qua BC.

`pair B=rotate(180,0)*A;` Trả về điểm B đối xứng với A qua O.

`pair B=rotate(k,0)*A;` Trả về điểm B là ảnh của A qua phép quay tâm O, góc k.

Một số điểm đặc biệt trong tam giác: trọng tâm, trực tâm, tâm ngoại tiếp, tâm nội tiếp, tâm bàng tiếp, tâm đường tròn *Mixtilinear*, tâm đường tròn 9 điểm, điểm *Gergonne*, điểm *Feuerbach*, điểm *Nagel*, ... tham khảo trong file **geo2d.asy** để biết tên hàm và các tham số.

Ví dụ sử dụng hàm *pair fepoint(pair A, pair B, pair C)* và *pair feexpoint(pair A, pair B, pair C)* để khai báo điểm *Feurbach*:

`pair Fe=fepoint(A,B,C);` Trả về điểm *Feuerbach* trong của tam giác $\triangle ABC$.

```
pair Fa=feexpoint(A,B,C);
```

Trả về điểm *Feuerbach* ngoài ứng với đỉnh A của tam giác $\triangle ABC$. Lưu ý có 3 điểm Fe ngoài, để khai báo Fb, Fc chỉ cần thay đổi tham số đầu tiên của hàm: pair Fb = feexpoint(B,C,A); pair Fc = feexpoint(C,A,B);

Để vẽ điểm có thể dùng lệnh **draw** hoặc **dot**:

```
pair A=(1.1,4);
dot(Label("$A$"),align=NW),A);
```

Khai báo điểm A, gán nhãn "A".

Tham số **align** để xác định vị trí của nhãn theo hướng bao gồm N (bắc), S (nam), E (đông), W (tây), NE (đông bắc), NW (tây bắc), SE (đông nam), SW (tây nam).

Có thể thêm giá trị vào **align** để thay đổi khoảng cách từ nhãn đến điểm. Có thể xác định kích thước của nhãn, kích thước và màu của điểm:

```
dot(Label("$A$"),align=2N,fontsize(8pt)),A,2bp);
```

Nhãn "A" của điểm A kích thước 2bp.

2.4 Đoạn thẳng, đường thẳng

Đoạn thẳng trong ASY kiểu *path*. Nó được khai báo như sau:

```
path pt= A--B;
```

Đoạn thẳng AB

ASY không định nghĩa đường thẳng nên sử dụng hàm *path lineex(pair A, pair B, real a=0.1, real b=a);* trong gói *geo2d.asy* để vẽ trong một số tình huống.

```
path LAB0=lineex(A,B);
```

Đoạn thẳng AB được kéo dài về hai đầu mút 0.1AB.

```
path LAB1=lineex(A,B,0.1,0.3);
```

Đoạn AB được kéo dài về đầu A 0.1AB, đầu B 0.3AB

```
pair N=parallel(M,A,B);
```

Trả về điểm N sao cho MN song song với AB (geo2d.asy).

```
pair N=perpendicular(M,A,B);
```

Trả về điểm N sao cho MN vuông góc với AB (geo2d.asy).

Cách sử dụng để vẽ sẽ được thể hiện trong các ví dụ tiếp theo.

```
draw(A--B);
```

Vẽ đoạn A-B.

```
draw(A--B^^X--Y);
```

Vẽ các đoạn AB, XY.

```
draw(A--B--C--cycle);
```

Vẽ tam giác $\triangle ABC$.

```
draw(lineex(A,B,0.1,0.3));
```

Vẽ đoạn AB được kéo dài về đầu A 0.1AB, đầu B 0.3AB.

```
draw("$a$",B--C,2bp+blue+dashed);
```

Vẽ đoạn BC kích thước 0.8bp màu xanh và gán tên là "a".

Ở đoạn mã cuối, lưu ý:

- Cách sử dụng hàm "lồng nhau" như trên để "tiết kiệm" việc khai báo biến. Mới ban đầu việc sử dụng có thể khó khăn, tuy nhiên khi dùng quen sẽ thấy tiện lợi vì đoạn mã trở nên gọn gàng hơn.
- *dashed* là kiểu của *path* (kiểu gạch ngang), có các kiểu như sau: *solid* (ngầm định), *dotted*, *dashed*, *longdashed*, *dashdotted* và *longdashdotted*. Có thể thay vào mã để hình dung về từng kiểu.

2.5 Đường tròn, cung tròn, E-líp

ASY sử dụng hàm *path circle(pair c, real r)* để định nghĩa đường tròn. Nó sẽ trả về một đường cong Bezier gần với đường tròn và nó dựa trên đường tròn đơn vị (unitcircle). Từ hàm này, trong gói *geo2d.asy* có thêm các hàm sau:

<code>path cOA=circle1p(O, A);</code>	Đường tròn tâm O, bán kính OA.
<code>path cAB=circle2p(A, B);</code>	Đường tròn đường kính AB.
<code>path cABC=circle3p(A, B, C);</code>	Đường tròn đi qua 3 điểm A, B, C.
<code>pair O=center3p(A, B, C);</code>	Tâm đường tròn đi qua 3 điểm A, B, C.
<code>path cIN=incircle(A, B, C);</code>	Đường tròn nội tiếp tam giác $\triangle ABC$.
<code>path I=incenter(A, B, C);</code>	Tâm nội tiếp tam giác $\triangle ABC$.

Tương tự như *circle*, hàm *path ellipse(pair c, real a, real b)* cũng dựa trên đường tròn đơn vị và phép biến hình tỷ lệ *scale*. ASY chuẩn cũng chỉ có hàm trên về ellipse. Nếu sử dụng về ellipse, parabola, hyperbola và các đường bậc hai nói chung, nên sử dụng gói *geometry.asy* được đề cập ở mục tiếp theo.

2.6 Giao điểm

Xác định giao điểm của các đối tượng trên rất quan trọng. ASY dùng hàm *intersectionpoint(path pt0, path pt1);* hoặc *intersectionpoints(path pt0, path pt1);* để xác định giao điểm của các path.

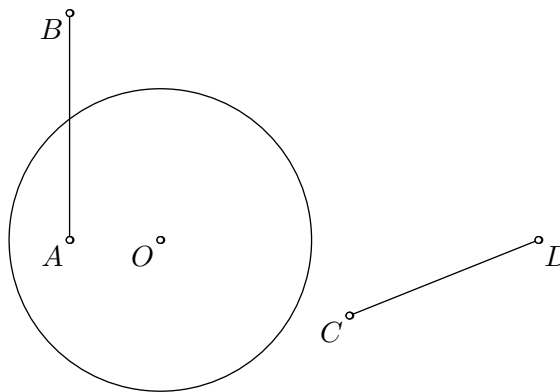
<code>pair I=intersectionpoint(pt0, pt1);</code>	Trả về giao điểm của path pt0, pt1.
<code>pair I[]=intersectionpoints(pt0, pt1);</code>	Trả về các giao điểm của path pt0, pt1.

Nếu hai path là hai đường thẳng thì có thể dùng hàm *pair extension(pair P, pair Q, pair p, pair q)*. Hàm này trả về giao điểm của đường thẳng PQ và đường thẳng pq.

<code>pair H=extension(A, B, X, Y);</code>	Trả về H là giao điểm của hai đường thẳng AB và XY.
--	---

Do *path* là đường có giới hạn nên việc xét giao đôi khi cần phải điều chỉnh.

Ví dụ 1. Cho đường thẳng AB, CD và đường tròn (O) như hình vẽ. Xác định giao điểm của đường thẳng AB và CD; AB, CD với (O).

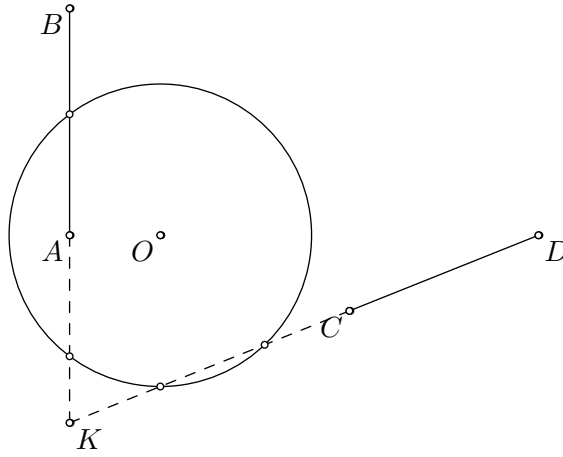


Ở ví dụ này nếu sử dụng:

<code>pair e[]=intersectionpoints(A--B, cO);</code>	Chỉ trả về 1 giao điểm của đoạn thẳng AB và (O).
<code>pair g[]=intersectionpoints(C--D, cO);</code>	Không có giao điểm của đường thẳng CD và (O).
<code>pair K=intersectionpoint(A--B, C--D);</code>	Không có giao điểm của đường thẳng AB và CD.

Khi đó cần sử dụng hàm *lineex* hoặc *extension* như sau:

<code>pair e[]=intersectionpoints(lineex(A, B, 1000), cO);</code>
<code>pair g[]=intersectionpoints(lineex(C, D, 1000), cO);</code>
<code>pair K=extension(A, B, C, D);</code>



Khi xét tiếp điểm của đường tròn với đường thẳng và đường tròn với đường tròn cũng cần lưu ý vì hàm *circle* trả về đường cong *Bezier* gần với đường tròn.

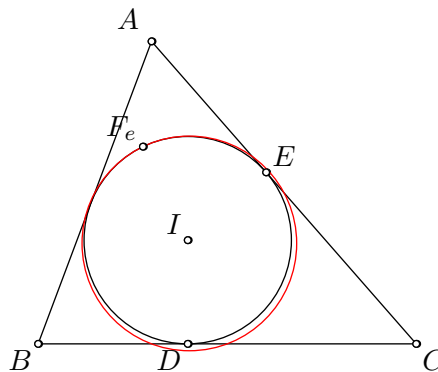
Ví dụ 2. Cho tam giác $\triangle ABC$ và đường tròn nội tiếp $\odot(I)$. Xác định tiếp điểm của $\odot(I)$ với các cạnh của tam giác $\triangle ABC$ và đường tròn 9 điểm.

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair A=(1.5,4); dot(Label("$A$"),align=2NW,A);
pair B=(0,0); dot(Label("$B$"),align=SW,B);
pair C=(5,0); dot(Label("$C$"),align=SE,C);
draw(A--B--C--A);
pair I=incenter(A,B,C); dot(Label("$I$"),align=NW,I);
path cIN=incircle(A,B,C); draw(cIN);
pair D=intersectionpoint(incircle(A,B,C),B--C); dot(Label("$D$"),align=SW,D);
pair E=intersectionpoint(cIN,A--C); dot(Label("$E$"),align=NE,E);
path cN9=n9circle(A,B,C); draw(cN9);
pair Fe=intersectionpoint(cIN,cN9); dot(Label("$F_e$"),align=NW,Fe);

dot(A--B--C--D--E--I--Fe,Fill(white));
```

Khi dịch, ta được kết quả như hình dưới:



Tuy nhiên nếu đổi tọa độ điểm A, ví dụ *pair A=(1.1,4)*; và dịch lại thì sẽ có thông báo “paths do not intersect”. Kiểm tra lỗi, ta sẽ thấy lỗi ở dòng khai báo điểm E hoặc ở dòng khai báo điểm Fe.

Hàm *incircle(A,B,C)* và *n9circle(A,B,C)* là hàm trong gói *geo2d.asy* và không phải nó lỗi. Như đã nói trên, hàm *path circle(pair c, real r)* thực chất là đường cong *Bezier* gần với đường tròn nên nó chưa chắc có giao điểm dạng tiếp điểm.

Tiếp điểm D luôn tồn tại do trong hàm *incircle(A,B,C)* sử dụng điểm D để lấy bán kính. Nếu đổi tọa độ điểm *A(1.1,4)* sẽ không báo lỗi như trên.

Như vậy khi muốn xác định tiếp điểm giữa đường tròn với đường thẳng và đường tròn với đường tròn, không nên dùng hàm *intersectionpoint* hoặc *intersectionpoints*. Sử dụng cách khác để xác định tiếp điểm như sau:

```
pair E=projection(I,C,A);
pair Fe=intersectionpoint(lineex(n9center(A,B,C),I,0,10000),cIN);
```

E được xác định là hình chiếu vuông góc của I lên CA. Hàm *lineex(N,I,0,10000)* trả về đoạn thẳng từ điểm N, kéo dài về phía I $10000 \cdot NI$. Tham khảo thêm cách xác định điểm Fe trong hàm *fepoint* của gói **geo2d.asy**.

2.7 Một số ví dụ

Các ví dụ dưới đây chúng tôi viết trên *TexStudio* và *Notepad++*. Khi dịch ở chế độ lệnh, có thể cần khai báo thêm header.

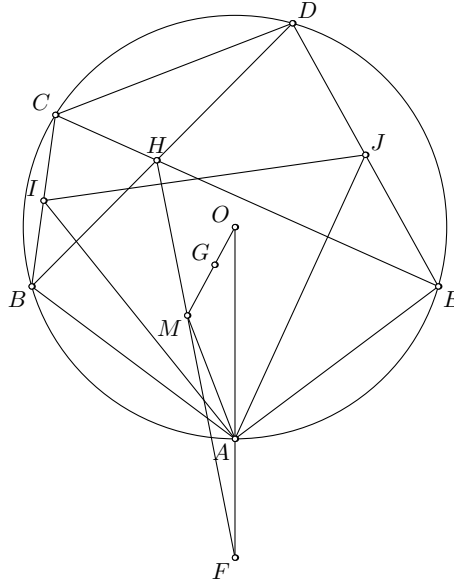
Việc copy mã từ tài liệu này để chạy có thể gây lỗi. Bạn đọc nên tự gõ các lệnh để có thể kiểm tra được lỗi, hoặc tham khảo mã theo link: <https://github.com/geometryvn/asymptote/tree/master/geo2d/examples>.

Ví dụ 1 (2019 China TST Test 1 Day 1 Q1). Cho ngũ giác $ABCDE$ nội tiếp đường tròn $\odot(O)$ có $AB = AE = CD$. Gọi F là trực tâm tam giác $\triangle ABE$, H là giao BD, CE . Gọi I, J là trung điểm BC, DE . Gọi G là trọng tâm $\triangle AIJ$. OG cắt FH tại M . Chứng minh $AM \perp CD$.

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair O=(0,0); dot(Label("$O$"),align=NW,O);
pair A=(0,-3.5); dot(Label("$A$"),align=SW,A);
real k=4.2;
pair B=intersectionpoints(circle1p(O,A),circle(A,k))[0];
dot(Label("$B$"),align=SW,B);
pair E=reflect(O,A)*B; dot(Label("$E$"),align=SE,E);
pair C=rotate(-122,O)*A; dot(Label("$C$"),align=NW,C);
pair D=intersectionpoints(circle1p(O,A),circle(C,k))[0];
dot(Label("$D$"),align=NE,D);
pair F=orthocenter(A,B,E); dot(Label("$F$"),align=SW,F);
pair H=extension(B,D,C,E); dot(Label("$H$"),align=N,H);
pair I=midpoint(B--C); dot(Label("$I$"),align=NW,I);
pair J=midpoint(D--E); dot(Label("$J$"),align=NE,J);
pair G=centroid(A,I,J); dot(Label("$G$"),align=NW,G);
pair M=extension(O,G,F,H); dot(Label("$M$"),align=SW,M);

draw(circle1p(O,A));
draw(A--B--C--D--E--A^^B--D--C--E--A--M^^A--I--J--A^^O--F^^O--M^^F--H);
dot(A^^B^^C^^D^^E^^F^^G^^H^^I^^J^^M^^O,Fill(white));
```



Ở ví dụ 1 cần lưu ý những điều sau:

- Muốn thay đổi kích thước đường tròn, thay đổi tọa độ điểm $A(0, -3.5)$.
- Muốn thay đổi vị trí các điểm B, C, D, E thì thay đổi giá trị k .
- Cách xác định điểm B là cách gọi các hàm lồng nhau. Để rõ ràng, có thể làm như sau:

```
pair [] b=intersectionpoints(circle1p(O,A),circle(A,k));
pair B=b[0];
```

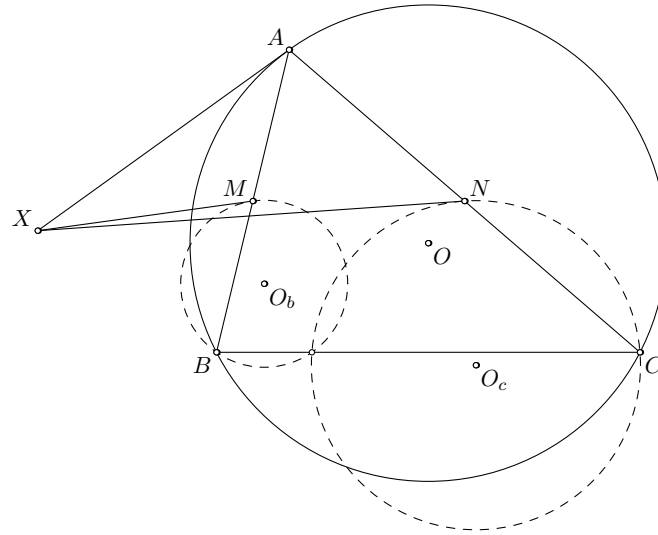
- Lưu ý sử dụng các phép biến hình: rotate, reflect.

Ví dụ 2 (USA TST 2019, Bài 1). Cho $\triangle ABC$ có M, N lần lượt là trung điểm AB, AC . Gọi $\odot(O)$ là đường tròn ngoại tiếp $\triangle ABC$. Gọi X là một điểm bất kỳ trên tiếp tuyến qua A của $\odot(O)$. Gọi $\odot\omega_B$ là đường tròn đi qua M, B và tiếp xúc với MX , $\odot\omega_C$ là đường tròn đi qua N, C và tiếp xúc với NX . Chứng minh rằng: $\odot\omega_B$ và $\odot\omega_C$ cắt nhau trên BC .

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair A=(1.5,5); dot(Label("$A$"),align=NW,A);
pair B=(0,0); dot(Label("$B$"),align=SW,B);
pair C=(7,0); dot(Label("$C$"),align=SE,C);
pair O=center3p(A,B,C); dot(Label("$O$"),align=SE,O);
path cO=circle3p(A,B,C); draw(cO);
pair M=midpoint(A--B); dot(Label("$M$"),align=NW,M);
pair N=midpoint(A--C); dot(Label("$N$"),align=NE,N);
pair X=scale(A,1.1)*rotate(-90,A)*O; dot(Label("$X$"),align=NW,X);
pair Ob=extension(midpoint(M--B),midline(M,B),M,rotate(90,M)*X);
dot(Label("$O_b$"),align=SE,Ob);
pair Oc=extension(midpoint(N--C),midline(N,C),N,rotate(90,N)*X);
dot(Label("$O_c$"),align=SE,Oc);
pair t[]=intersectionpoints(circle1p(Ob,B),circle1p(Oc,C));

draw(circle1p(Ob,B)^^circle1p(Oc,C),dashed);
draw(A--B--C--A^^A--X);
dot(A^^B^^C^^M^^N^^O^^X^^Ob^^Oc^^t[1],Fill(white));
```



Ở ví dụ 2, cần lưu ý: Điểm X trên tiếp tuyến tại A của (O) được xác định bằng hai phép biến hình quay và vị tự. Thay đổi giá trị k để thay đổi vị trí điểm X. Có thể viết như sau:

```
pair x=rotate(-90,A)*O;
pair X=scale(A,1.1)*x;
```

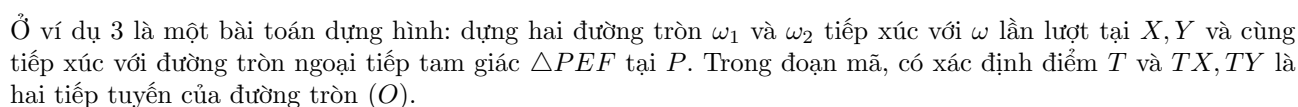
Ví dụ 3 (Iran TST 2018, Test 3 Day 2). Cho tứ giác $ABCD$ nội tiếp đường tròn ω . $P \equiv AC \cap BD$. E, F lần lượt nằm trên các cạnh AB, CD sao cho $\angle APE = \angle DPF$. Hai đường tròn ω_1 và ω_2 tiếp xúc với ω lần lượt tại X, Y và cùng tiếp xúc với đường tròn ngoại tiếp tam giác $\triangle PEF$ tại P . Chứng minh $\frac{EX}{EY} = \frac{FX}{FY}$.

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair A=(0.2,3.5); dot(Label("$A$"),align=NW,A);
pair D=(0,0); dot(Label("$D$"),align=SW,D);
pair C=(7,0); dot(Label("$C$"),align=SE,C);
pair O=center3p(A,C,D); dot(Label("$O$"),align=SE,O);
path c0=circle3p(A,C,D); draw(c0);
pair B=rotate(-70,O)*A; dot(Label("$B$"),align=NW,B);
pair P=extension(A,C,B,D); dot(Label("$P$"),align=NW,P);
real angle=70;
pair E=extension(A,B,P,rotate(-angle,P)*A); dot(Label("$E$"),align=NW,E);
pair F=extension(C,D,P,rotate(angle,P)*D); dot(Label("$F$"),align=SW,F);

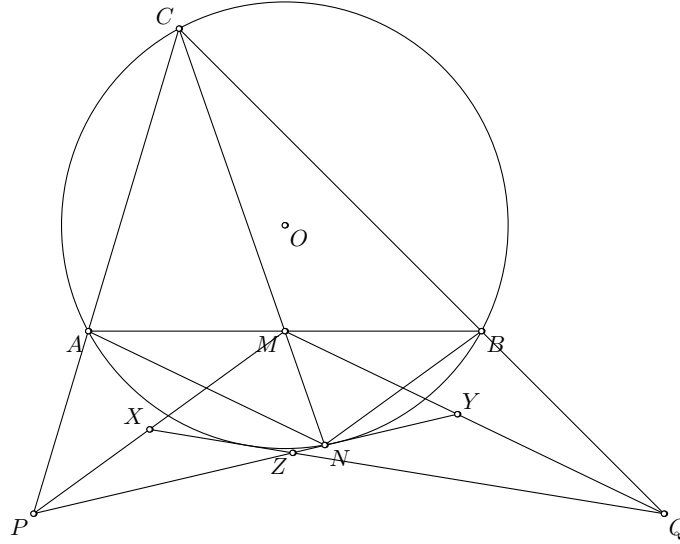
path cP=circle3p(P,E,F); draw(cP,dashed);
pair T=extension(E,F,P,rotate(90,P)*center3p(P,E,F));
//dot(Label("$T$"),align=SW,T);
pair X=intersectionpoints(c0,circle1p(T,P))[0]; dot(Label("$X$"),align=NE,X);
pair Y=reflect(O,T)*X; dot(Label("$Y$"),align=NW,Y);
pair O1=extension(O,X,P,center3p(P,E,F)); draw(circle1p(O1,P));
pair O2=extension(O,Y,P,center3p(P,E,F)); draw(circle1p(O2,P));

draw(A--B--C--D--A^^A--C^^B--D^^P--E^^P--F);
dot(A^^B^^C^^D^^E^^F^^O^^P^^X^^Y,Fill(white));
```



Ví dụ 4 (International Zhaautykov Olympiad 2019 - P3). Cho tam giác $\triangle ABC$. Đường trung tuyến CM cắt lại đường tròn $\odot(ABC)$ tại N . Điểm P và Q lần lượt nằm trên tia CA và CB , thỏa $PM \parallel BN$ và $QM \parallel AN$. Điểm X và Y lần lượt nằm trên đoạn PM và QM , thỏa mãn PY và QX là tiếp tuyến của đường tròn (ABC) . Gọi $Z = PY \cap QX$. Chứng minh rằng tứ giác $MXZY$ ngoại tiếp.

20



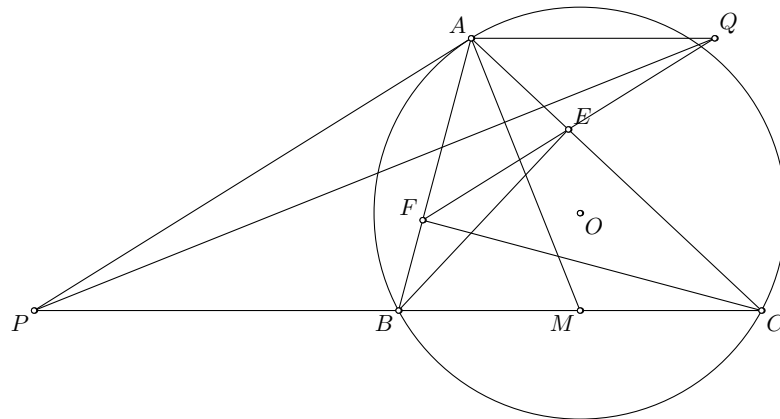
Ở ví dụ 4 là một cách khác để xác định điểm N thay vì xét giao của CM và đường tròn (O) . Do CM đã cắt (O) tại điểm C nên sử dụng cách này để phân biệt giao điểm thứ hai với C . Cách này sẽ hiệu quả khi lập trình các hàm tìm giao.

Ví dụ 5 (Polish MO Finals 2018, Problem 5). Cho tam giác nhọn $\triangle ABC$ ($AB < AC$) nội tiếp đường tròn $\odot(O)$ có BE, CF là hai đường cao. Tiếp tuyến tại A của $\odot(O)$ cắt BC tại P . Đường thẳng qua A song song với BC cắt EF tại Q . Gọi M là trung điểm của BC . Chứng minh $AM \perp PQ$.

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair A=(1.2,4.5); dot(Label("$A$"),align=NW,A);
pair B=(0,0); dot(Label("$B$"),align=SW,B);
pair C=(6,0); dot(Label("$C$"),align=SE,C);
pair O=center3p(A,B,C); dot(Label("$O$"),align=SE,O);
path cO=circle3p(A,B,C); draw(cO);
pair E=projection(B,C,A); dot(Label("$E$"),align=NE,E);
pair F=projection(C,A,B); dot(Label("$F$"),align=NW,F);
pair P=extension(B,C,A,rotate(90,A)*O); dot(Label("$P$"),align=SW,P);
pair Q=extension(E,F,A,parallel(A,B,C)); dot(Label("$Q$"),align=NE,Q);
pair M=midpoint(B--C); dot(Label("$M$"),align=SW,M);

draw(A--B--C--A^^B--E^^C--F^^A--M^^A--P^^P--B^^A--Q^^Q--F^^P--Q);
dot(A^^B^^C^^E^^F^^M^^O^^P^^Q,Fill(white));
```



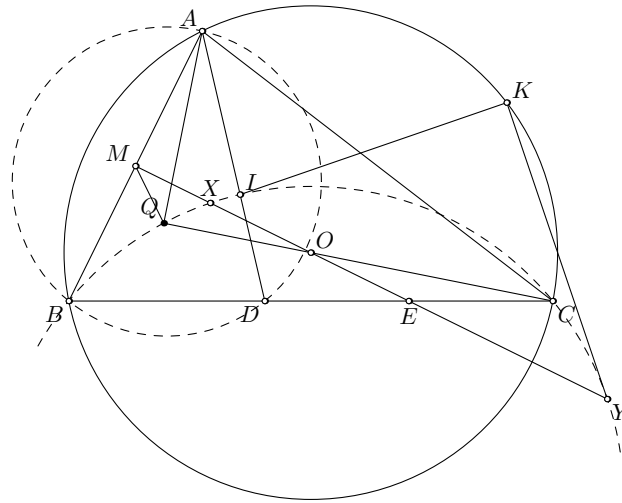
Ví dụ 6 (USA TST 2018). Cho tam giác $\triangle ABC$ nhọn, I là tâm đường tròn nội tiếp, O là tâm đường tròn ngoại tiếp và đường tròn ngoại tiếp Γ . Gọi M là trung điểm đoạn AB . Tia AI cắt BC tại D . Gọi ω và γ lần lượt là các đường tròn ngoại tiếp các tam giác BIC và BAD . Đường thẳng MO cắt ω tại X và Y , trong khi đó

đường thẳng CO cắt ω tại C và Q . Giả sử rằng Q nằm trong tam giác ABC và $\angle AQM = \angle ACB$. Biết rằng $\angle BCA \neq 60^\circ$. Chứng minh rằng các tiếp tuyến từ X, Y của ω và từ A, D của γ đồng quy.

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair B=(0,0); dot(Label("$B$",align=SW),B);
pair C=(8,0); dot(Label("$C$",align=SE),C);
pair Ma=midpoint(B--C);
pair O=interp(Ma,rotate(-90,Ma)*B,0.2); dot(Label("$O$",align=NE),O);
draw(circle1p(O,B));
pair L=midpoint(arc(O,B,C));
pair E=intersectionpoint((Ma--C),circle2p(O,L)); dot(Label("$E$",align=S),E);
pair K=reflect(O,E)*L; dot(Label("$K$",align=NE),K);
pair A=reflect(O,E)*B; dot(Label("$A$",align=NW),A);
pair M=midpoint(A--B); dot(Label("$M$",align=NW),M);
pair X=intersectionpoint((M--O),circle1p(L,B)); dot(Label("$X$",align=N),X);
pair Y=intersectionpoint(lineex(O,E,0,4),circle1p(L,B));
dot(Label("$Y$",align=SE),Y);
pair Q=reflect(L,reflect(O,C)*L)*C; dot(Label("$Q$",align=NW),Q);
pair I=incenter(A,B,C); dot(Label("$I$",align=NE),I);
pair D=extension(A,I,B,C); dot(Label("$D$",align=SW),D);

draw(arcex(L,Y,B),dashed);
draw(circle3p(A,B,D),dashed);
draw(A--B--C--cycle^^A--D--M--Y^^K--I^^K--Y^^C--Q^^A--Q^^Q--M);
dot(A^^B^^C^^D^^E^^I^^K^^M^^O^^X^^Y,Fill(white));
```



Ở ví dụ 6, hàm $midpoint(arc(O,B,C))$ trả về điểm chính giữa cung nhỏ \widehat{BC} của đường tròn (O) . Để lấy điểm chính giữa cung lớn \widehat{BC} , sử dụng $midpoint(arc(O,C,B))$.

Ví dụ này cũng là một bài toán dựng hình: dựng tam giác $\triangle ABC$ sao cho $\angle AQM = \angle ACB$.

Ví dụ 7 (Saudi Arabia TST for IMO 2018, P4). Cho tam giác nhọn không cân $\triangle ABC$. Gọi M, N, P lần lượt là trung điểm BC, CA, AB . Gọi d_1 là đường đi qua M và vuông góc với đường phân giác trong góc $\angle A$ của $\triangle ABC$. Định nghĩa tương tự với d_2 và d_3 . Gọi $D = d_2 \cap d_3, E = d_3 \cap d_1, F = d_1 \cap d_2$. Gọi I, H lần lượt là tâm nội tiếp và trực tâm của $\triangle ABC$. Chứng minh rằng tâm của đường tròn $\odot(DEF)$ là trung điểm IH .

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

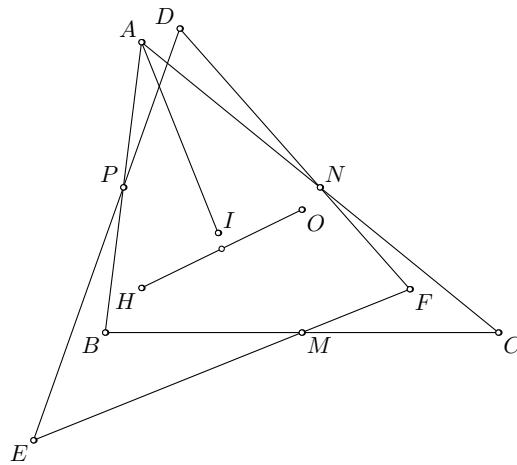
pair A=(0.6,4.8); dot(Label("$A$",align=NW),A);
pair B=(0,0); dot(Label("$B$",align=SW),B);
pair C=(6.5,0); dot(Label("$C$",align=SE),C);
```

```

pair I=incenter(A,B,C); dot(Label("$I$",align=NE),I);
pair O=center3p(A,B,C); dot(Label("$O$",align=SE),O);
pair H=orthocenter(A,B,C); dot(Label("$H$",align=SW),H);
pair M=midpoint(B--C); dot(Label("$M$",align=SE),M);
pair N=midpoint(C--A); dot(Label("$N$",align=NE),N);
pair P=midpoint(A--B); dot(Label("$P$",align=NW),P);
pair D=extension(N,reflect(B,I)*N,P,reflect(C,I)*P);
dot(Label("$D$",align=NW),D);
pair E=extension(M,reflect(A,I)*M,P,reflect(C,I)*P);
dot(Label("$E$",align=SW),E);
pair F=extension(N,reflect(B,I)*N,M,reflect(A,I)*M);
dot(Label("$F$",align=SE),F);
pair T=midpoint(O--H);

draw(A--B--C--cycle^^D--E^^E--F^^D--F^^H--O^^A--I);
dot(A^^B^^C^^D^^E^^F^^H^^I^^M^^N^^P^^O^^T,Fill(white));

```



Ví dụ 8. (Đề chọn đội tuyển Cần Thơ 2019-2020, Bài 4). Cho tam giác $\triangle ABC$ có đường tròn nội tiếp (I) lần lượt tiếp xúc với CA, AB tại E, F . Gọi D là điểm trên BC sao cho $\angle AID = 90^\circ$. Đường tròn bàng tiếp ứng với đỉnh A, B, C lần lượt tiếp xúc với BC, CA, AB tại P, N, M . Gọi X là giao điểm của (ABC) và (AEF) ($X \neq A$). Chứng minh rằng:

- A, D, X thẳng hàng và AD tiếp xúc với (AMN) .
- Nếu tứ giác $AMPN$ nội tiếp được trong một đường tròn thì AD là tiếp tuyến của (DMN) .

```

import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

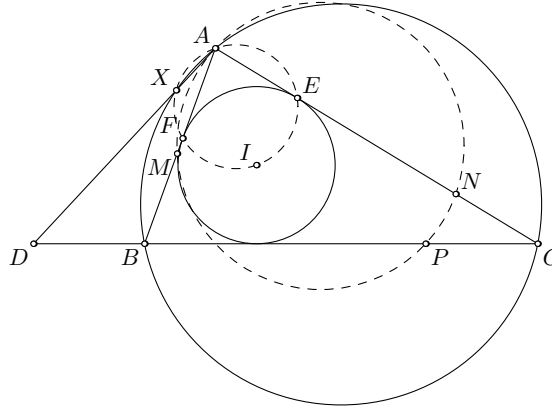
pair B=(0,0); dot(Label("$B$",align=SW),B);
pair C=(6.5,0); dot(Label("$C$",align=SE),C);
pair M=midpoint(B--C);
pair O=interp(M,rotate(-90,M)*B,0.2); draw(circle1p(O,B));
pair Ma=midpoint(arc(O,B,C));
pair B1=rotate(180,O)*B;
pair C1=rotate(180,O)*C;
pair I=intersectionpoints(lineex(B1,C1),circle1p(Ma,B))[1];
dot(Label("$I$",align=NW),I);
pair P=rotate(180,O)*I; dot(Label("$P$",align=SE),P);
pair A=reflect(O,reflect(Ma,I)*O)*Ma; dot(Label("$A$",align=NW),A);
pair E=projection(I,C,A); dot(Label("$E$",align=NE),E);
pair F=projection(I,A,B); dot(Label("$F$",align=NW),F);
pair N=projection(excenter(B,C,A),C,A); dot(Label("$N$",align=NE),N);
pair M=projection(excenter(C,A,B),A,B); dot(Label("$M$",align=SW),M);
pair D=extension(B,C,I,rotate(90,I)*A); dot(Label("$D$",align=SW),D);
pair X=reflect(O,reflect(A,D)*O)*A; dot(Label("$X$",align=NW),X);

```

```

draw(incircle(A,B,C));
draw(circle3p(A,E,F)^^circle3p(A,M,N),dashed);
draw(A--B--C--cycle^^D--A^^D--B);
dot(A^^B^^C^^D^^E^^F^^I^^M^^N^^P^^X,Fill(white));

```



Ví dụ 8 thực ra là bài **USA TST 2019, P6**. Câu b là một bài toán dựng hình: dựng tam giác $\triangle ABC$ sao cho đường tròn (MNP) đi qua A .

Ví dụ 9 (Đề chọn đội tuyển PTNK 2019, Bài 4). Cho tam giác $\triangle ABC$ nhọn nội tiếp đường tròn $\odot(O)$ với B, C cố định và A thay đổi trên cung lớn \widehat{BC} . Các đường tròn bàng tiếp góc A, B, C lần lượt tiếp xúc với BC, CA, AB tại D, E, F .

a) Gọi $L \neq A$ là giao điểm của (ABE) và (ACF) . Chứng minh AL luôn đi qua điểm cố định.

b) (BCF) cắt (BAD) tại M, B và (CAD) cắt (CBE) tại N, C . Gọi K, I, J theo thứ tự là trung điểm của AD, BE, CF . Chứng minh KL, IM, JN đồng qui.

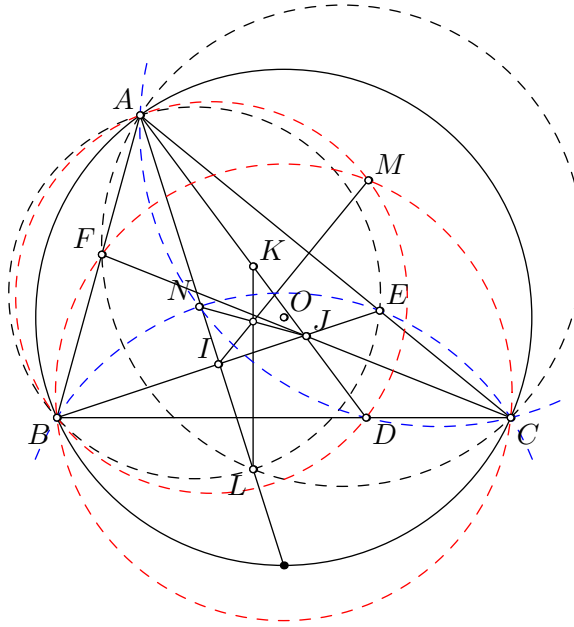
```

import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair A=(1.1,4); dot(Label("$A$"),align=NW,A);
pair B=(0,0); dot(Label("$B$"),align=SW,B);
pair C=(6,0); dot(Label("$C$"),align=SE,C);
pair O=center3p(A,B,C); dot(Label("$O$"),align=NE,O);
draw(circle3p(A,B,C));
pair D=projection(excenter(A,B,C),B,C); dot(Label("$D$"),align=SE,D);
pair E=projection(excenter(B,C,A),C,A); dot(Label("$E$"),align=NE,E);
pair F=projection(excenter(C,A,B),A,B); dot(Label("$F$"),align=NW,F);
pair L=reflect(center3p(A,B,E),center3p(A,C,F))*A; dot(Label("$L$"),align=SW,L);
pair L1=midpoint(arc(O,B,C)); dot(L1);
pair M=reflect(center3p(B,C,F),center3p(B,A,D))*B; dot(Label("$M$"),align=NE,M);
pair N=reflect(center3p(C,A,D),center3p(C,B,E))*C; dot(Label("$N$"),align=NW,N);
pair K=midpoint(A--D); dot(Label("$K$"),align=NE,K);
pair I=midpoint(B--E); dot(Label("$I$"),align=NW,I);
pair J=midpoint(C--F); dot(Label("$J$"),align=NE,J);
pair T=extension(I,M,J,N);

draw(A--B--C--A^^A--L1^^A--D^^B--E^^C--F^^K--L^^I--M^^J--N);
draw(circle3p(A,B,E)^^circle3p(A,C,F),dashed);
draw(circle3p(B,C,F)^^circle3p(B,A,D),dashed);
draw(circle3p(C,A,D)^^circle3p(C,B,E),dashed);
dot(A^^B^^C^^D^^E^^F^^I^^J^^K^^L^^M^^N^^O^^T,Fill(white));

```

Ví dụ 10 (Bài toán đề nghị cho IGO 2019 - Trần Quân). Cho tam giác $\triangle ABC$ ($AB < AC$) nội tiếp đường tròn $\odot(O)$ có I là tâm nội tiếp. Gọi S là điểm trên BC sao cho SA tiếp xúc với $\odot(O)$. Gọi J là điểm trên AI sao cho trung điểm của các đoạn thẳng AS, BI và CJ cùng nằm trên một đường thẳng. Vẽ đường tròn $\odot(J)$ tiếp xúc với AB, AC . Tiếp tuyến song song với BC của $\odot(J)$ lần lượt cắt AB, AC tại P, Q (J, A nằm khác phía so với PQ). Chứng minh đường tròn $\odot(BPI)$ và đường tròn $\odot(CQJ)$ tiếp xúc với nhau.

```
import geo2d;
unitsize(1cm);
defaultpen(fontsize(11pt));

pair A=(0.8,4.5); dot(Label("$A$"),align=NW,A);
pair B=(0,0); dot(Label("$B$"),align=SW,B);
pair C=(7,0); dot(Label("$C$"),align=SE,C);
path c0=circle3p(A,B,C); draw(c0);
pair O=center3p(A,B,C); dot(Label("$O$"),align=NE,O);
pair I=incenter(A,B,C); dot(Label("$I$"),align=NW,I);
pair S=extension(B,C,A,rotate(90,A)*O); dot(Label("$S$"),align=SW,S);
pair mAS=midpoint(A--S), mBI=midpoint(B--I), mCA=midpoint(C--A);
pair mCJ=extension(mAS,mBI,mCA,parallel(mCA,A,I));
pair J=extension(A,I,C,mCJ); dot(Label("$J$"),align=SW,J);
pair e=projection(J,C,A), f=projection(J,A,B);
path cJ=circle1p(J,e); //draw(cJ,dashed);
pair T=intersectionpoint(cJ,J--reflect(B,C)*J); //dot(Label("$T$"),align=SE,T);
pair P=extension(A,B,T,parallel(T,B,C)); dot(Label("$P$"),align=NW,P);
pair Q=extension(A,C,T,parallel(T,B,C)); dot(Label("$Q$"),align=NE,Q);

draw(circle3p(B,P,I)^^circle3p(C,Q,J),blue);
draw(arcex(J,e,f,20),dashed);
draw(A--B--C--cycle^^A--S^^A--J^^S--B--P--Q--B--I^^C--J^^f--B^^lineex(mAS,mCJ));
dot(A^^B^^C^^I^^J^^O^^P^^Q^^S^^T^^e^^f^^mAS^^mBI^^mCJ,Fill(white));
```



- 26