

# VẼ HÌNH TRONG L<sup>A</sup>T<sub>E</sub>X VỚI METAPOST

Phan Tấn Phú  
phantanphu@gmail.com

Ngày 07 – 02 – 2009

## Mục lục

<b>1</b>	<b>Mở đầu</b>	<b>2</b>
<b>2</b>	<b>Làm thử</b>	<b>2</b>
<b>3</b>	<b>Quy trình vẽ hình bằng METAPOST</b>	<b>3</b>
<b>4</b>	<b>Cấu trúc của file mp</b>	<b>3</b>
<b>5</b>	<b>Thực hành</b>	<b>3</b>
5.1	Tam giác . . . . .	3
5.2	Hình bình hành . . . . .	4
5.3	Trọng tâm tam giác, giao điểm hai đường thẳng . . . . .	5
5.4	Phép quay, tam giác vuông cân . . . . .	5
5.5	Tam giác đều . . . . .	5
5.6	Đường cao, hình chiếu vuông góc . . . . .	5
5.7	Đường phân giác . . . . .	8
5.8	Sử dụng macro . . . . .	8
5.9	Giao ba đường cao, ba đường phân giác, ba đường trung trực . . . . .	9
5.10	Đường tròn ngoại tiếp, nội tiếp tam giác . . . . .	10
5.11	Các tùy chọn nét vẽ . . . . .	10
5.12	Sử dụng màu . . . . .	10
5.13	Đường . . . . .	10
5.14	Tô màu . . . . .	12
5.15	Các phép biến hình . . . . .	12
5.16	Sử dụng tiếng Việt trong METAPOST . . . . .	12
5.17	Vòng lặp for . . . . .	13
5.18	Điều kiện if . . . . .	13
<b>6</b>	<b>Tài liệu tham khảo</b>	<b>13</b>

# 1 Mở đầu

Ta có thể vẽ hình bằng các phần mềm chuyên dụng rồi chèn file hình vẽ được vào  $\text{T}_\text{E}\text{X}$ . Các phần mềm vẽ hình này thì dễ sử dụng nhưng hình xuất ra thường là dạng bitmap nên khi chèn vào  $\text{T}_\text{E}\text{X}$  cho chất lượng bản in không cao, file xuất ra có dung lượng lớn và sẽ bị bể hạt khi phóng lớn ảnh. Việc vẽ hình trực tiếp trong  $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$  thì có thể sử dụng gói `pstricks` hoặc `METAPOST` hoặc có thể là những gói khác. Ưu điểm của cách làm này là cho chất lượng bản in cao, dung lượng file nhỏ, nhưng ngược lại thì cần phải thuộc lệnh. Việc chọn phương án nào tùy vào yêu cầu của hình cần vẽ và kinh nghiệm sử dụng  $\text{T}_\text{E}\text{X}$  của mỗi người. Ở tài liệu này tôi xin trình bày cách sử dụng `METAPOST`.

Tài liệu này được trình bày dưới dạng các bài tập thực hành. Qua từng bài thực hành tôi sẽ giải thích ý nghĩa từng lệnh được sử dụng. Lệnh nào đã giải thích ở bài trước thì không giải thích lại. Do đó, khi đọc tài liệu này, nếu là người mới bắt đầu sử dụng `METAPOST` tôi khuyên bạn nên đọc theo thứ tự từ đầu đến cuối.

## 2 Làm thử

### Bước 1

Soạn một file có phần mở rộng là `*.mp`, chẳng hạn `tamgiac.mp` và lưu ở một thư mục nào đó, chẳng hạn `C:\lamthu` với nội dung như sau:

```
beginfig(1)
u:=1cm;
pair a, b, c;
a:=(0,0);
b:=(3u,0);
c:=(u,2u);
draw a--b--c--cycle;
label.lft(btex  $A$  etex, a);
label.rt(btex  $B$  etex, b);
label.top(btex  $C$  etex, c);
endfig;
end;
```

### Bước 2

Mở command prompt, di chuyển đến thư mục `C:\lamthu` và đánh lệnh `mpost tamgiac.mp`. Nếu hệ thống `MiKTeX` của bạn chưa có gói `METAPOST` thì sẽ có thông báo yêu cầu cài đặt `METAPOST`. Sau khi biên dịch thành công, vào thư mục `lamthu` bạn sẽ thấy file `tamgiac.1` được tạo ra.

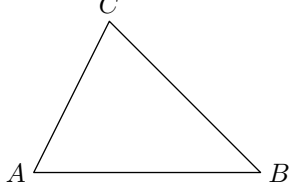
Nếu bạn sử dụng trình soạn thảo là `TexMaker` thì không cần phải vào dos đánh lệnh `mpost` mà ở cửa sổ soạn thảo file `mp` có thể dùng chức năng `MPost` trong menu `Tools`.

### Bước 3

Soạn một file `tex` với nội dung như sau:

---

Hình đầu tiên của tôi:



---

Hình 1: Làm thử

```
\documentclass[12pt, a4paper]{article}
\usepackage[utf8]{vietnam}
\usepackage{graphicx}
\begin{document}
Hình đầu tiên của tôi:\\
\convertMPtoPDF{tamgiac.1}{1}{1}
\end{document}
```

Biên dịch sang PDF bằng pdflatex ta sẽ được kết quả như hình 1

Lệnh `\convertMPtoPDF` dùng để chèn vào T<sub>E</sub>X file `tamgiac.1` vừa tạo ra. Để sử dụng được lệnh này bạn cần nạp gói `graphicx` hoặc gói `xcolor`. Chú ý rằng lệnh này chỉ hoạt động trong pdflatex, không hoạt động khi biên dịch sang dvi.

### 3 Quy trình vẽ hình bằng METAPOST

METAPOST là một gói của MiKTeX. Để thực hiện vẽ hình bằng METAPOST và chèn vào T<sub>E</sub>X bạn cần phải thực hiện qua các công đoạn sau:

- Soạn một file và lưu với phần mở rộng `*.mp`
- Biên dịch file `*.mp` bằng lệnh `mpost name.mp` để thu được file hình `*.n`
- Chèn file hình `*.n` vào file `*.tex` bằng lệnh `\convertMPtoPDF{name.n}{num}{num}`

Hai tham số `num` là hai số dùng để điều chỉnh độ phóng đại của hình được chèn.

### 4 Cấu trúc của file mp

Nội dung của file `mp` phải có các cặp từ khoá `beginfig(n)...` `endfig;`. Phần giữa của cặp từ khoá này (phần ba chấm) chứa các lệnh khai báo đối tượng và các lệnh vẽ hình. Kết thúc file là từ khoá `end;`. Trong một file có thể có một hoặc nhiều cặp từ khoá trên nhưng tham số `n` trong mỗi cặp từ khoá đó phải khác nhau. Giả sử file `mp` đang soạn là `hinh.mp`, có bao nhiêu cặp từ khoá trên trong file này thì khi `mpost` sẽ xuất ra bấy nhiêu file `hinh.n`. Các tham số `n` trong mỗi cặp từ khoá chính là phần mở rộng của các file hình được xuất ra và phải là các số tự nhiên `0, 1, 2, ...`

## 5 Thực hành

### 5.1 Tam giác

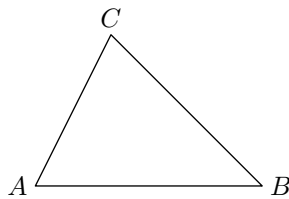
Trong hình 2, từ khoá `numeric` dùng để khai báo biến kiểu số thực. Ở ví dụ này có thể không cần đến lệnh `numeric u`; vì khi thực hiện phép gán giá trị cho một biến mới chưa

---

```

beginfig(1)
numeric u; % có thể không cần lệnh này, hãy xem giải thích
u:=1cm; % tạo một hằng số độ dài
pair a, b, c; % khai báo ba điểm
a:=(0,0); % gán tọa độ
b:=(3u,0);
c:=(u,2u);
draw a--b--c--cycle; % vẽ đường gấp khúc a--b--c khép kín về a
label.lft(btex $A$ etex, a); % tạo nhãn có tên A tại điểm a
label.rt(btex $B$ etex, b);
label.top(btex $C$ etex, c);
endfig;
end;

```




---

Hình 2: Tam giác




---

Hình 3: Lệnh label

từng khai báo trước đó thì METAPOST mặc định hiểu đó là một biến kiểu số thực. Ta dùng một hằng số độ dài  $u$  để tiện cho sau này khi muốn thay đổi kích thước của hình.

Ta có một kiểu dữ liệu khác nữa là kiểu `pair`, hiểu nôm na là một bộ hai tọa độ gồm hoành độ và tung độ. Kiểu này có thể xem như là điểm hoặc cũng có thể xem như vectơ. Ta có thể thực hiện các phép toán cộng, trừ trên biến kiểu này.

Lệnh `label.top(btex some-text etex, P );` dùng để tạo nhãn có tên `some-text` ở vị trí trên của điểm  $P$ . Cặp từ khóa `btex` và `etex` là bắt buộc. Nội dung của `some-text` có thể chứa công thức toán. Điểm  $P$  phải là một biến kiểu `pair`.

Vị trí trên, dưới, trái, phải, ... của nhãn được quy định như hình 3. Nếu muốn tạo nhãn ngay chính giữa điểm  $P$  thì dùng lệnh `label(btex some-text etex, P );`, nếu muốn vẽ một dấu chấm thì thay `label` bằng `dotlabel`

Lệnh `draw a-b` dùng để vẽ đoạn thẳng nối hai điểm  $a$  và  $b$ . Nếu muốn vẽ đường khép kín thì dùng thêm `cycle`. Nếu muốn vẽ mũi tên từ  $a$  đến  $b$  ta dùng `drawarrow a-b`.

## 5.2 Hình bình hành

Trong hình 4, hình bình hành này được xác định bởi các tham số đầu vào là vị trí điểm  $A$  (hay `a[0]`) và hai vectơ  $\vec{v}$ ,  $\vec{w}$  (hay `v`, `w`). Muốn thay đổi vị trí của hình bình hành thì ta thay đổi tọa độ `p[0]`, còn để thay đổi hình dạng ta điều chỉnh tọa độ `v`, `w`.

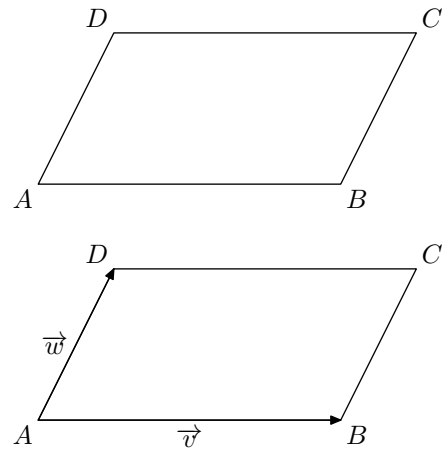
---

```

beginfig(1)
u:=1cm;
pair a[]; % khai báo một mảng các điểm
pair v, w; % khai báo hai vectơ
v:=(4u,0);
w:=(u,2u);
a[0]:=(0,0);
a[1]:=a[0]+v;
a[2]:=a[1]+w;
a[3]:=a[0]+w;
draw a[0]--a[1]--a[2]--a[3]--cycle;
label.llft(btex $A$ etex, a[0]);
label.lrt(btex $B$ etex, a[1]);
label.urc(btex $C$ etex, a[2]);
label.ulft(btex $D$ etex, a[3]);
endfig;
end;

```

---



Hình 4: Hình bình hành (mã lệnh tương ứng với hình trên, hình dưới chỉ để giải thích)

### 5.3 Trọng tâm tam giác, giao điểm hai đường thẳng

Trong hình 5, lệnh `m:=1/2[b,c]` dùng để gán cho `m` (điểm  $M$ ) là một điểm trên đường thẳng `bc` (đường thẳng  $BC$ ) sao cho  $\overrightarrow{BM} = \frac{1}{2}\overrightarrow{BC}$ . Như vậy  $M$  là trung điểm của  $BC$ . Tổng quát hơn `t[a,m]` biểu thị cho một điểm nào đó trên đường thẳng nối hai điểm `a` và `m` sao cho vectơ từ `a` đến điểm đó gấp `t` lần vectơ từ `a` đến `m`. Còn `whatever[a,m]` để chỉ một điểm nào đó trên đường thẳng `am`, do đó `g:=whatever[a,m] = whatever[b,n]`; gán cho `g` là giao điểm của hai đường thẳng `am` và `bn`. Ta hay dùng lệnh này để xác định giao điểm của hai đường thẳng.

### 5.4 Phép quay, tam giác vuông cân

Trong hình 6, lệnh `B rotatedaround(A,90)` là quay `B` quanh `A` một góc  $90^\circ$ . Nếu tâm của phép quay là gốc tọa độ thì dùng lệnh `rotated 90`.

### 5.5 Tam giác đều

Trong hình 7, `dir(-30)` là một điểm (pair, có thể xem là một vectơ) nằm trên đường tròn đơn vị có tọa độ  $(\cos 30^\circ, \sin 30^\circ)$ . Tương tự `r*dir(-30)` là một điểm (pair) trên đường tròn tâm là gốc tọa độ, bán kính `r` có tọa độ  $(r \cos 30^\circ, r \sin 30^\circ)$ . Như vậy, hàm `dir(góc)` trả về một pair như đã nói ở trên.

Một hàm ngược của hàm `dir` là hàm `angle(pair)` tác động lên một pair (hay một vectơ) trả về một góc có số đo bằng độ là góc tạo bởi vectơ (pair) đó và trục hoành. Lệnh này được sử dụng để vẽ đường phân giác của góc sẽ trình bày sau.

### 5.6 Đường cao, hình chiếu vuông góc

Trong hình 8, `(C-B) rotated 90` là vectơ  $\overrightarrow{BC}$  quay một góc  $90^\circ$  (gọi đó là vectơ  $\vec{n}$ ). Tiếp theo `A+((C-B) rotated 90)` là một điểm (tạm gọi là  $M$ ) trên đường thẳng qua `A` và

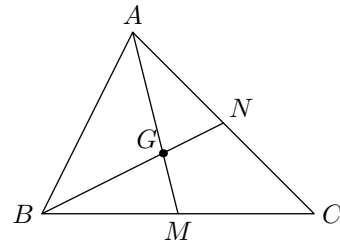
---

```

beginfig(1)
u:=1.2cm;
pair a, b, c, m, n, g;
a:=(u,2u);
b:=(0,0);
c:=(3u,0);
m:=1/2[b,c];
n:=1/2[a,c];
g:=whatever [a,m] = whatever [b,n];
draw a--b--c--cycle;
draw a--m;
draw b--n;
label.top(btex "$A$" etex, a);
label.lft(btex "$B$" etex, b);
label.rft(btex "$C$" etex, c);
label.bot(btex "$M$" etex, m);
label.urft(btex "$N$" etex, n);
dotlabel.ulft(btex "$G$" etex, g);
endfig;
end

```

---



Hình 5: Trọng tâm tam giác

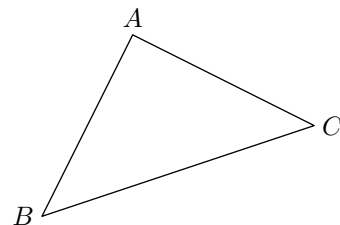
---

```

beginfig(1)
u:=1.2cm;
pair A, B, C;
A:=(0,u);
B:=(-u,-u);
C:=B rotatedaround (A,90);
draw A--B--C--cycle;
label.top(btex "$A$" etex, A);
label.lft(btex "$B$" etex, B);
label.rft(btex "$C$" etex, C);
endfig;
end

```

---



Hình 6: Phép quay, tam giác vuông cân

---

```
beginfig(1)
```

```
u:=1.2cm;
```

```
numeric r;
```

```
r:=3u;
```

```
pair A, B, C;
```

```
A:=r*dir(-30);
```

```
B:=r*dir(90);
```

```
C:=r*dir(210);
```

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

```
label.rt(btex  $A$  etex, A);
```

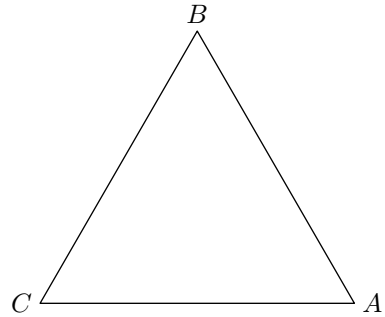
```
label.top(btex  $B$  etex, B);
```

```
label.lft(btex  $C$  etex, C);
```

```
endfig;
```

```
end
```

---



Hình 7: Tam giác đều

---

```
beginfig(1)
```

```
u:=1.5cm;
```

```
pair A, B, C, H;
```

```
A:=(u,2u);
```

```
B:=(0,0);
```

```
C:=(3u,0);
```

```
H:=whatever[B, C]
```

```
  = whatever[A, A+((C-B) rotated 90)];
```

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

```
draw A--H;
```

```
label.top(btex  $A$  etex, A);
```

```
label.lft(btex  $B$  etex, B);
```

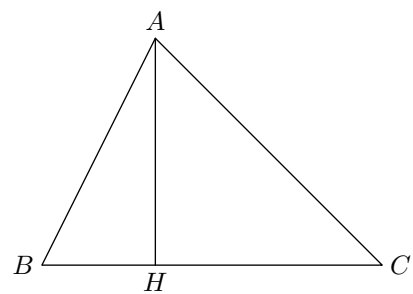
```
label.rt(btex  $C$  etex, C);
```

```
label.bot(btex  $H$  etex, H);
```

```
endfig;
```

```
end
```

---



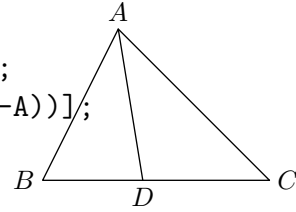
Hình 8: Đường cao

---

```

beginfig(1)
u:=1cm;
pair A, B, C, D;
A:=(u,2u);
B:=(0,0);
C:=(3u,0);
D:=whatever[B, C]
  = A+whatever*dir(.5angle(B-A)+.5angle(C-A));
%hoặc=whatever[A,A+dir(.5angle(B-A)+.5angle(C-A))];
draw A--B--C--cycle;
draw A--D;
label.top(btex  $A$  etex, A);
label.lft(btex  $B$  etex, B);
label.rt(btex  $C$  etex, C);
label.bot(btex  $D$  etex, D);
endfig;
end

```




---

Hình 9: Đường phân giác

nhận  $\vec{n}$  làm vectơ chỉ phương. Từ đó `whatever[A,A+((C-B) rotated 90)]` là một điểm nào đó trên đường thẳng  $AM$ . Như vậy lệnh `H:=whatever[B,C]=whatever[A,A+((C-B) rotated 90)]`; nhằm gán cho  $H$  là giao điểm của hai đường thẳng  $BC$  và  $AM$ . Vậy  $H$  là hình chiếu của điểm  $A$  lên đường thẳng  $BC$ .

Đoạn mã `whatever[A,A+((C-B) rotated 90)]` có thể được viết lại là `A+whatever*((C-B) rotated 90)` với ý nghĩa tương đương và cùng được dùng để mô tả một điểm nào đó trên đường thẳng đi qua  $A$  và có vectơ chỉ phương là  $\vec{n}$ .

## 5.7 Đường phân giác

Trong hình 9, `angle(B-A)` là số đo góc  $\alpha$  tạo bởi vectơ  $\overrightarrow{AB}$  và trục hoành, còn `angle(C-A)` là số đo góc  $\beta$  tạo bởi vectơ  $\overrightarrow{AC}$  và trục hoành. Do đó `.5angle(B-A)+.5angle(C-A)` là góc  $\gamma = \frac{\alpha+\beta}{2}$ . `dir(.5angle(B-A)+.5angle(C-A))` là một vectơ  $\vec{n}$  có hướng tạo với chiều dương trục hoành một góc  $\gamma$ . Từ đó, `A+whatever*dir(.5angle(B-A)+.5angle(C-A))` là một điểm nào đó trên đường thẳng qua  $A$  và có vectơ chỉ phương là  $\vec{n}$ . Nhờ vậy mà ta xác định được chân phân giác  $D$  kẻ từ đỉnh  $A$  của tam giác  $ABC$ .

## 5.8 Sử dụng macro

Quay lại hình 8, ta có thể định nghĩa thủ tục xác định chân đường cao của tam giác (hay hình chiếu vuông góc của một điểm lên một đường thẳng đi qua hai điểm khác) để sử dụng lại nhiều lần như hình 10.

Ta có thể soạn một file `macro.mp` chứa các định nghĩa sẵn. Muốn sử dụng các macro đã định nghĩa ta chỉ cần nạp file `macro.mp` bằng lệnh `input macro.mp`; trong file `hinh.mp` đang vẽ.

Để bắt đầu định nghĩa một macro ta dùng từ khoá `def` hoặc `vardef`, các tham số



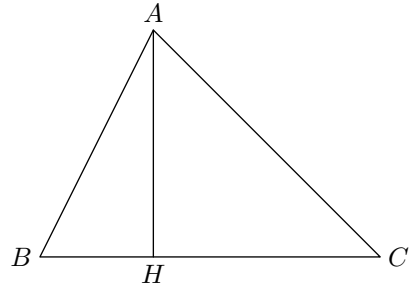
---

```

beginfig(1)
% định nghĩa macro
def hìnhchieu(expr m,a,b,h) =
    h = m + whatever*(b-a) rotated 90;
    h = whatever [a,b];
enddef; % kết thúc định nghĩa
u:=1.5cm;
pair A, B, C, H;
A:=(u,2u);
B:=(0,0);
C:=(3u,0);
hìnhchieu(A, B, C, H); % sử dụng macro
draw A--B--C--cycle;
draw A--H;
label.top(btex "$A$ etex, A);
label.lft(btex "$B$ etex, B);
label.rft(btex "$C$ etex, C);
label.bot(btex "$H$ etex, H);
endfig;
end

```

---



Hình 10: Macro hình chiếu

được liệt kê sau `expr`, các lệnh được viết sau dấu `=` và kết thúc macro bằng `enddef`;

Ở hình 8, khi sử dụng macro xác định hình chiếu đã định nghĩa bằng lệnh `hìnhchieu(A, B, C, H)`; thì đã thực hiện phép gán cho `H` là hình chiếu của `A` lên đường thẳng qua hai điểm `B` và `C`.

Tương tự macro xác định chân đường phân giác trong của tam giác được viết như sau:

```

def chaphangiac(expr m, a, b, h) =
    h=whatever [a,b];
    h=m+whatever*dir(.5angle(a-m)+.5angle(b-m));
enddef;

```

## 5.9 Giao ba đường cao, ba đường phân giác, ba đường trung trực

Sau đây là các macro xác định giao ba đường cao, ba đường phân giác, ba đường trung trực của tam giác. Ta có thể lưu chúng trong một file để sử dụng nhiều lần.

```

def giaobaduongcao (expr a, b, c, h) =
    h=a+whatever*((b-c) rotated 90);
    h=b+whatever*((c-a) rotated 90);
enddef;

def giaobaphangiac(expr a, b, c, d) =
    d=a+whatever*dir(.5angle(b-a)+.5angle(c-a));

```

```

d=b+whatever*dir(.5angle(c-b)+.5angle(a-b));
enddef;

def giaobatrungtruc (expr a, b, c, m) =
  m = .5[b,c] + whatever * ((b-c) rotated 90);
  m = .5[c,a] + whatever * ((c-a) rotated 90);
enddef;

```

## 5.10 Đường tròn ngoại tiếp, nội tiếp tam giác

Với các macro đã tạo sẵn, ta có thể vẽ được đường tròn nội tiếp, ngoại tiếp tam giác như hình 11. METAPOST có sẵn hàm `abs(pair)`, trong đó `pair` là đối số, hàm trả về một số thực là độ dài của vectơ `pair`. Lệnh `draw fullcircle scaled 2R shifted I`; vẽ một đường tròn có đường kính  $2R$  và có tâm là  $I$ . Nếu viết `draw fullcircle scaled d`; thì sẽ vẽ đường tròn đường kính  $d$  tâm tại gốc tọa độ.

## 5.11 Các tùy chọn nét vẽ

Ta có thể vẽ đường nét liền, nét đứt và có thể điều chỉnh độ đậm của nét vẽ như hình 12.

## 5.12 Sử dụng màu

Mỗi màu của METAPOST là một bộ ba số  $(r, g, b)$ . Trong đó  $r, g, b$  là các số thực trong đoạn  $[0; 1]$ . Một số màu được định nghĩa sẵn như `red`, `green`, `blue`, `white`, `yellow`, ... Ba màu cơ bản là `red` hay  $(1, 0, 0)$ ; `green` hay  $(0, 1, 0)$ ; `blue` hay  $(0, 0, 1)$ . Ta có thể thực hiện các phép toán cộng, trừ các màu và nhân một số thực  $t$  thuộc đoạn  $[0; 1]$  với các màu, từ đó mà tạo ra được rất nhiều màu khác.

## 5.13 Đường

Ta đã biết qua các kiểu dữ liệu `numeric` và `pair`. Một kiểu dữ liệu nữa là `path`. Một `path` là tập hợp các `pair`. Dạng đơn giản nhất của `path` là một điểm. Biến kiểu `path` có thể dùng để mô tả một đường cong hoặc một đường thẳng, đoạn thẳng.

Ta có thể tạo ra `path` từ các `pair` bằng các toán tử `--` hoặc `..`. Ví dụ, `A--B` là một `path` nối hai `pair`  $A$  và  $B$  bằng một đoạn thẳng, còn `A..B..C` là một `path` nối ba `pair`  $A, B, C$  bằng một đường cong tròn. METAPOST tự động xác định đường cong tròn này.

Thay vì viết

```
draw A--B--C;
```

Ta có thể viết

```
path p;
p:=A--B--C;
draw p;
```

Và sau này có thể sử dụng lại `p` thay cho `A-B-C`.

---

```

beginfig(1)
def giaobatrungtruc (expr a, b, c, m) =
  m = .5[b,c] + whatever * ((b-c) rotated 90);
  m = .5[c,a] + whatever * ((c-a) rotated 90);
enddef;

```

```

def giaobaphangiac(expr a, b, c, d)=
  d=a+whatever*dir(.5angle(b-a)+.5angle(c-a));
  d=b+whatever*dir(.5angle(c-b)+.5angle(a-b));
enddef;

```

```

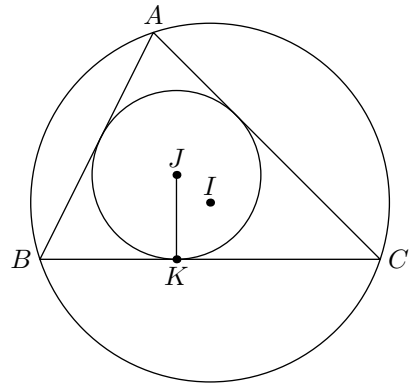
def hinhchieu(expr m,a,b,h) =
  h = m + whatever*(b-a) rotated 90;
  h = whatever [a,b];
enddef;

```

```

u:=1.5cm;
pair A, B, C, I, J, K;
A:=(u,2u);
B:=(0,0);
C:=(3u,0);
giaobatrungtruc(A, B, C, I);
giaobaphangiac(A, B, C, J);
hinhchieu(J, B, C, K);
R:=abs(A-I);
r:=abs(J-K);
draw A--B--C--cycle;
draw J--K;
draw fullcircle scaled 2R shifted I;
draw fullcircle scaled 2r shifted J;
label.top(btex "$A$" etex, A);
label.lft(btex "$B$" etex, B);
label.rt(btex "$C$" etex, C);
dotlabel.top(btex "$I$" etex, I);
dotlabel.top(btex "$J$" etex, J);
dotlabel.bot(btex "$K$" etex, K);
endfig;
end

```




---

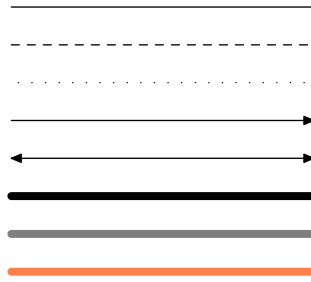
Hình 11: Đường tròn ngoại tiếp, nội tiếp tam giác

---

```

beginfig(1)
u:=.5cm;
draw (0,0)--(8u,0);
draw (0,-u)--(8u,-u) dashed evenly; % đường nét đứt
draw (0,-2u)--(8u,-2u) dashed withdots; % đường "chấm chấm"
drawarrow (0,-3u)--(8u,-3u); % mũi tên
drawdblarrow (0,-4u)--(8u,-4u); % mũi tên hai đầu
draw (0,-5u)--(8u,-5u) withpen pencircle scaled 3pt; % nét đậm
draw (0,-6u)--(8u,-6u) withpen pencircle scaled 3pt withcolor .5white; % màu
draw (0,-7u)--(8u,-7u) withpen pencircle scaled 3pt withcolor (1, .5, .3); % màu
endfig;
end;

```




---

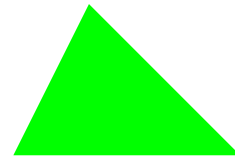
Hình 12: Các tùy chọn nét vẽ

---

```

beginfig(1)
u:=1cm;
pair A, B, C;
A:=(u,2u);
B:=(0,0);
C:=(3u,0);
fill A--B--C--cycle withcolor green;
endfig;
end

```




---

Hình 13: Tô màu miền tam giác

## 5.14 Tô màu

Hình 13 là ví dụ về tô màu một miền tam giác. Miền được tô màu phải là một miền kín. Miền kín này có thể được tạo ra từ các `path`. Cú pháp của lệnh tô màu là `fill mien withcolor mau`. Trong đó `mien` là miền kín cần tô, `mau` là tên của màu hoặc bộ ba số dùng để xác định màu. Ở hình 13, `A--B--C--cycle` là một `path` kín.

## 5.15 Các phép biến hình

## 5.16 Sử dụng tiếng Việt trong METAPOST

Tiếng Việt phải được đặt trong lệnh `label`. Để sử dụng Tiếng Việt, ta cần có thêm các lệnh sau trong phần mở đầu của file `*.mp`

```
verbatimtext
```

```
\documentclass{article}
\usepackage[utf8]{vietnam}
\begin{document}
etex
```

Khi biên dịch file \*.mp, ta dùng lệnh

```
mpost -tex=latex *.mp
```

trong đó, \* là tên file đang soạn. Khi gặp đoạn lệnh trên, .....

## 5.17 Vòng lặp for

## 5.18 Điều kiện if

# 6 Tài liệu tham khảo

- [1 ] John D. Hobby, *A user's manual for MetaPost*, AT&T Bell Laboratories Computing Science Technical Report 162, 1992.
- [2 ] André Heck, *Learning MetaPost by doing*, AMSTEL Institute, Universiteit van Amsterdam, 2005.
- [3 ] <http://tex.loria.fr/prod-graph/zoonekynd/metapost/metapost.html>, *Métapost examples*.