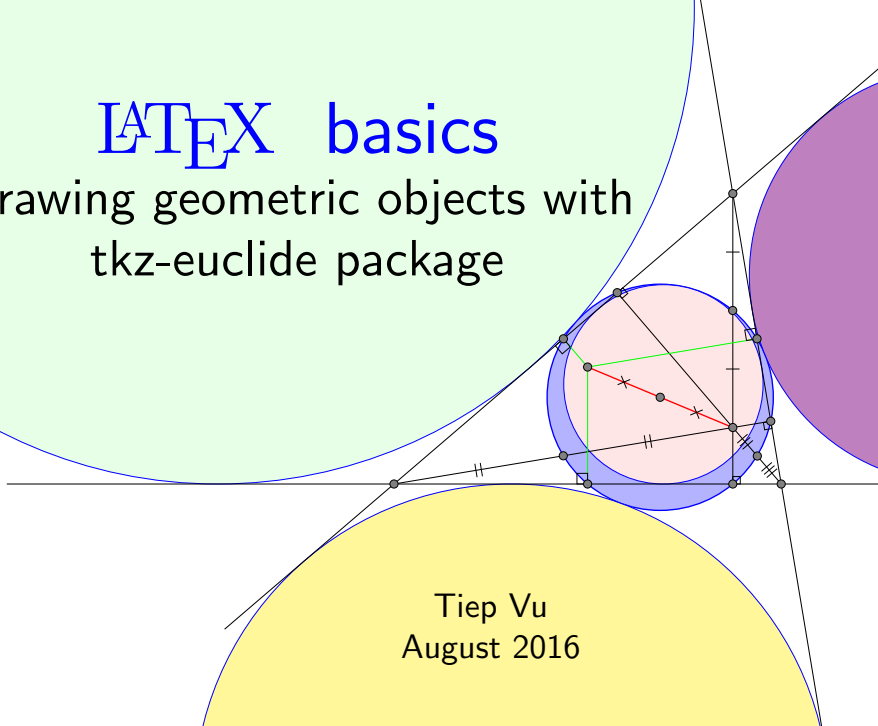


# L<sup>A</sup>T<sub>E</sub>X basics

Drawing geometric objects with  
tkz-euclide package



Tiep Vu  
August 2016

# Outline

## ① Points

- Points in Cartesian coordinate system
- Polar coordinate system
- Relative coordinates

## ② Lines, Segments, Rays

- Connect points
- Intersections
- Orthogonal and Parallel

## ③ Angles

- Specifying angle
- Labeling and Markers
- Angle bisector

## ④ Circles

- Drawing Circles
- Circle Intersection
- Circle and Tangents

## ⑤ Triangles

- Drawing triangles
- Centroid, Orthocenter, Circumcircle, Inscribed Circle

## tkz-euclide

## • documentclass

```
\documentclass{standalone}  
% \documentclass{article}
```

## • package

```
\usepackage{tkz-euclide}  
\usetkzobj{all}
```

## • main body

```
\begin{document}  
% \begin{figure}  
\begin{tikzpicture}  
  ....  
\end{tikzpicture}  
% \end{figure}  
\end{document}
```

•  $C$  $B$   
• $A$   
•

•

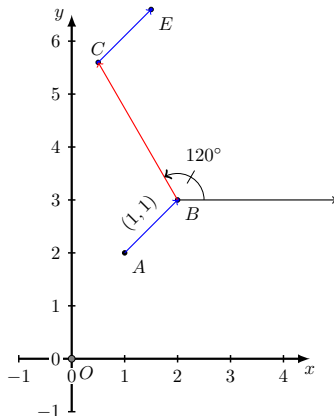
•  $D$ 

See [tkz-euclide examples here](#).

Learn L<sup>A</sup>T<sub>E</sub>X

Drawing geometric objects with tkz-euclide package

## Points



Tiep Vu Huu

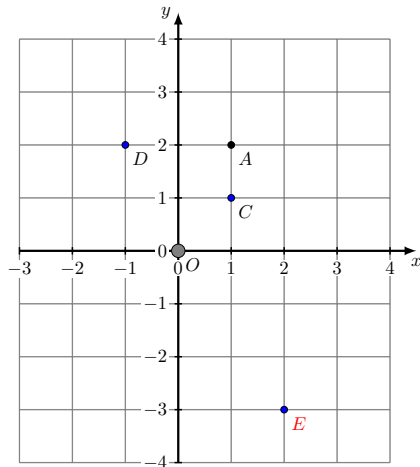
# Points in Cartesian coordinate system

```

%% def_point_Cartesian.tex
...
\begin{document}
\begin{tikzpicture}
  ...
  % Cartesian coordinate (x,y)
  \coordinate (O) at (0, 0);
  \coordinate (A) at (1, 2);
  % using tkz-euclide
  \tkzDefPoint(1,1){C}
  \tkzDefPoints{-1/2/D, 2/-3/E}
  ...
\end{tikzpicture}
\end{document}

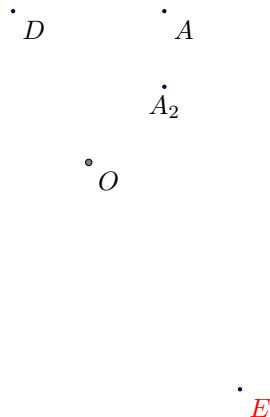
```

[See more here.](#)



# Drawing and Labeling points

```
%% draw_label_points.tex
...
\begin{tikzpicture}
...
% drawing points
\tkzDrawPoints(O)
\tkzDrawPoints[size=3, fill=black] (A)
\tkzDrawPoints[size=3, fill=blue] (D,E)
% labeling points
\tkzLabelPoints(O,A,D)
\tkzLabelPoints[red] (E)
% labeling in mathmode
\tkzDefPoint[label = below:$A_2$] (1,1) {A2}
\tkzDrawPoints[size=3, fill=blue] (A2)
\end{tikzpicture}
```



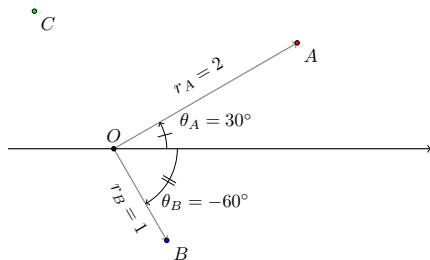
# Points in polar coordinate system

```

%% def_point_polar.tex
...
\begin{document}
\begin{tikzpicture}
...
\coordinate (O) at (0, 0);
% polar coordinate, (theta:r)
\coordinate (A) at (30:2);
\coordinate (B) at (-60:1);

\tkzDefPoint(120:1.5){C}
...
\end{tikzpicture}
\end{document}

```

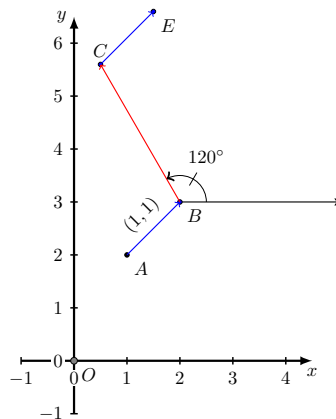


# Relative coordinates

```

%% relative_coordinate.tex
% ...
\usepackage{calc}
\begin{document}
\begin{tikzpicture}
% ...
\coordinate (O) at (0, 0);
\coordinate (A) at (1, 2);
\coordinate (B) at ($(A) + (1, 1)$);
\coordinate (C) at ($(B) + (120:3)$);
\coordinate (E) at ($(C) + (B) - (A)$);
% ...
\end{tikzpicture}
\end{document}

```

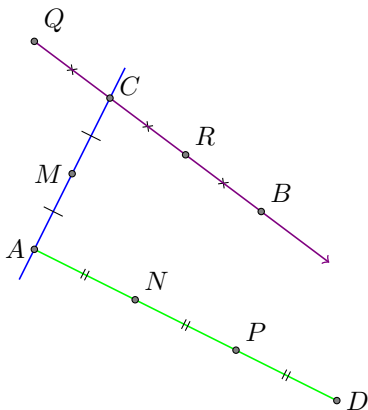




Learn L<sup>A</sup>T<sub>E</sub>X

Drawing geometric objects with tkz-euclide package

# Lines, Segments, Rays



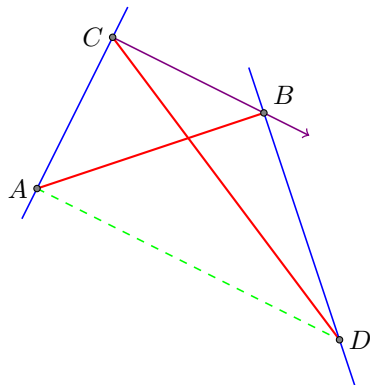
Tiep Vu Huu

# Line, Segment, Ray, connecting two points

```

%% line_segment_ray.tex
\begin{tikzpicture}
  % ...
  % drawing red segments
  \tkzDrawSegments[red, thick](A,B C,D)
  % drawing dashed green segment
  \tkzDrawSegments[green,dashed](A,D)
  % drawing blue line
  \tkzDrawLines[draw=blue](A,C B,D)
  % array with arrow
  \tkzDrawLines[add = 0 and 0.3, ...
    draw=violet, arrows=-->](C,B)
  % ...
\end{tikzpicture}

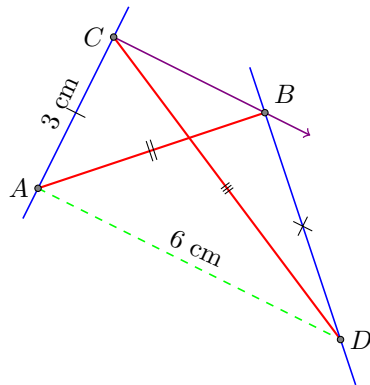
```



[See more here.](#)

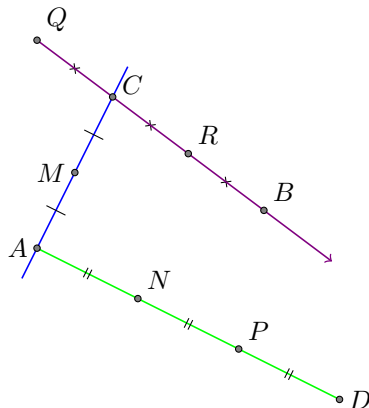
# Labeling and Markers

```
%% lsr_label_marker.tex
\begin{tikzpicture}
% ...
% labeling
\tkzLabelSegment[above=1pt, ...
  rotate=65](A,C){3 cm}
\tkzLabelSegment[above=0pt, ...
  rotate=-25](A,D){6 cm}
% markers
\tkzMarkSegment[mark=|](A,C)
\tkzMarkSegment[mark=||](A,B)
\tkzMarkSegment[mark=|||, size = ...
  2](C,D)
\tkzMarkSegment[mark=x](B,D)
% ...
\end{tikzpicture}
```



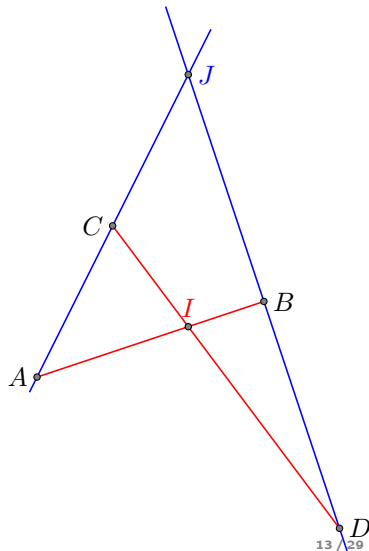
# Define a point between two points for a given length ratio

```
%% ratio_point.tex
\begin{tikzpicture}
  % ...
  % middle points
  \tkzDefMidPoint(A,C)\tkzGetPoint{M}
  % others
  \coordinate (N) at ($ (A) !1/3! (D) $);
  \coordinate (P) at ($ (A) !2/3! (D) $);
  \coordinate (Q) at ($ (C) !-1/2! (B) $);
  \coordinate (R) at ($ (C) !1/2! (B) $);
  % ...
\end{tikzpicture}
```



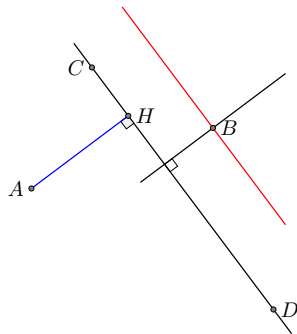
# Intersection of two lines

```
%% intersection1.tex
...
\begin{tikzpicture}
  %% defining points
  % InterLL for intersection of ...
  %   'Line' and 'Line'
  \tkzInterLL(A,B)(C,D) \tkzGetPoint{I}
  \tkzInterLL(A,C)(B,D) \tkzGetPoint{J}
  %% showing points
\end{tikzpicture}
```



# Orthogonal and Parallel lines

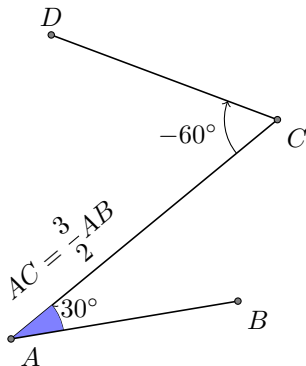
```
% orthogonal_parallel.tex
...
\begin{tikzpicture}
  % ...
  %% orthogonal
  \tkzDefPointBy[projection=onto C—D] (A)
  \tkzGetPoint{H}
  \tkzDefLine[orthogonal=through B] (C,D)
  \tkzGetPoint{K}
  %% parallel
  \tkzDefLine[parallel=through B] (C,D)
  \tkzDrawLine[draw = red, add = .5 and ...
    -.6] (B,tkzPointResult)
  % ...
\end{tikzpicture}
```



Learn L<sup>A</sup>T<sub>E</sub>X

Drawing geometric objects with tkz-euclide package

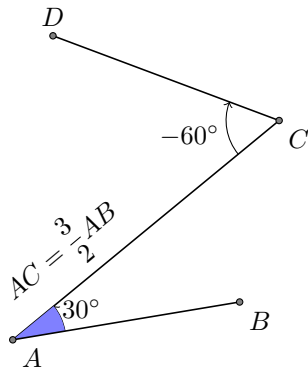
## Angles



Tiep Vu Huu

# Angles

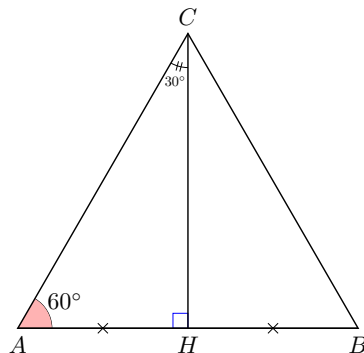
```
%% angle.tex
...
\begin{tikzpicture}
  % ...
  \coordinate (C) at ($(A)!1.5!30:(B)$);
  \coordinate (D) at ($(C)!.7!-60:(A)$);
  % ...
\end{tikzpicture}
```





# Labeling and Markers

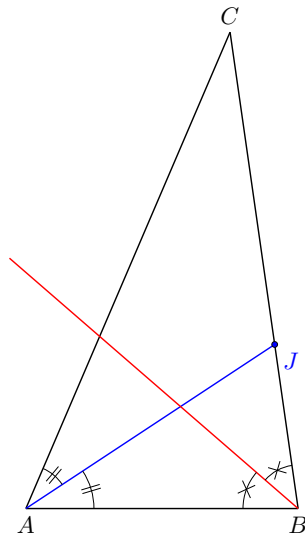
```
%% angle_label_marks.tex
...
\begin{tikzpicture}
  % ...
  % marker
  \tkzMarkAngle[size = .5, fill = ...
    red!30] (B,A,C)
  \tkzMarkAngle[size = .5, mark = ...
    ||,mksize=2] (A,C,H)
  % labeling
  \tkzLabelAngle[pos=.8] (B,A,C){ ...
    $60^\circ$}
  \tkzLabelAngle[pos=1.2,scale = ...
    .6] (A,C,H){$30^\circ$}
  % right angle marker
  \tkzMarkRightAngle[draw =blue] (A,H,C)
  % ...
\end{tikzpicture}
```



# Angle Bisector

```
%% angle_bisector.tex
...
\begin{tikzpicture}
  % ...
  \tkzDefLine[bisector] (C,B,A)
  \tkzGetPoint{i}

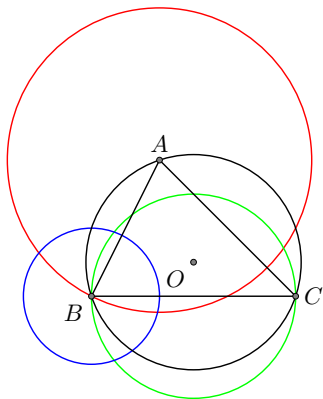
  \tkzDrawBisector[draw=blue] (C,A,B)
  \tkzGetPoint{J}
  % ...
\end{tikzpicture}
```



Learn L<sup>A</sup>T<sub>E</sub>X

Drawing geometric objects with tkz-euclide package

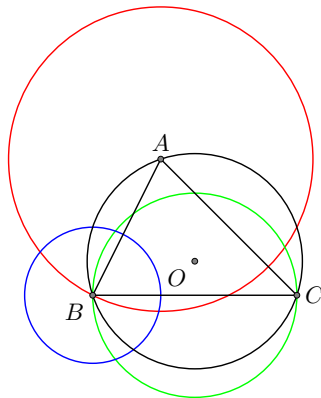
# Circles



Tiep Vu Huu

# Drawing Circles

```
% circle_1.tex
...
\begin{tikzpicture}
  % ...
  % center A, passing B
  \tkzDrawCircle[draw = red](A,B)
  % diameter BC
  \tkzDrawCircle[diameter, draw = ...
    green](B,C)
  % center B, radius 1 cm
  \tkzDrawCircle[R, draw = blue](B, ...
    1 cm)
  % passing A, B, C
  \tkzDrawCircle[circum](A,B,C)
  % get its center
  \tkzCircumCenter(A,B,C)\tkzGetPoint{O}
  % ...
\end{tikzpicture}
```

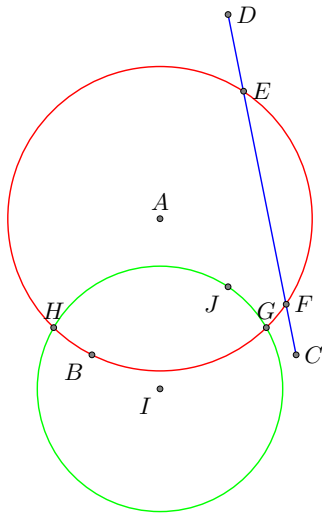


[See More Examples here.](#)

# Circle Intersection

```
% circle_intersection.tex
...
\begin{tikzpicture}
  % ...
  % Line-Circle intersection
  \tkzInterLC(C,D) (A,B)
  \tkzGetPoints{E}{F}
  % Circle-Circle intersection
  \tkzInterCC(A,B) (I,J)
  \tkzGetPoints{G}{H}
  % ...
\end{tikzpicture}
```

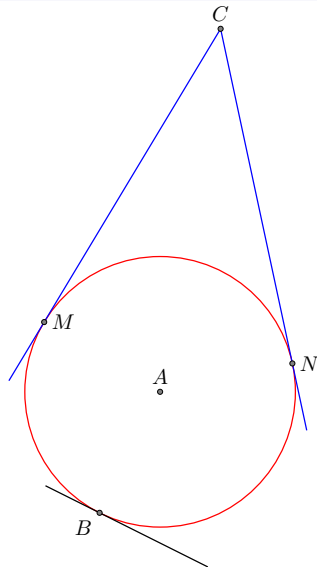
[See More Examples here.](#)



# Circle and Tangents

```
%% circle_tangent.tex
...
\begin{tikzpicture}
  % ...
  \tkzDrawCircle[draw = red] (A,B)
  % from a point on the circle
  \tkzTangent[at=B] (A) \tkzGetPoint{h}
  % from a point outside the circle
  \tkzTangent[from=C] (A,B)
  \tkzGetPoints{M}{N}
  % ...
\end{tikzpicture}
```

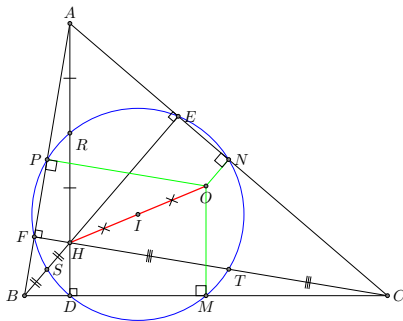
[See More Examples here.](#)



Learn L<sup>A</sup>T<sub>E</sub>X

Drawing geometric objects with tkz-euclide package

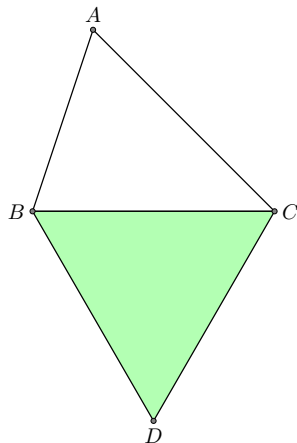
# Triangles



Tiep Vu Huu

# Drawing triangles

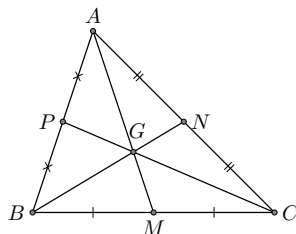
```
%% triangle_1.tex
...
\begin{tikzpicture}
  % ...
  % connecting 3 points
  \tkzDrawPolygon(A,B,C)
  % equilateral triangles
  \tkzDefTriangle[equilateral](C,B)
  \tkzGetPoint{D}
  \tkzDrawPolygon[fill=green!30](B,C,D)
  % ...
\end{tikzpicture}
```





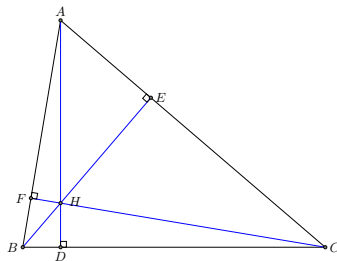
# Centroid

```
%% triangle_centroid.tex
...
\begin{tikzpicture}
  % ...
  % get centroid
  \tkzCentroid(A,B,C)\tkzGetPoint{G}
  % drawing median lines
  \tkzDrawLines[add = 0 and 1/2](A,G ...
    B,G C,G)
  % ...
\end{tikzpicture}
```



# Orthocenter

```
% triangle_orthocenter.tex
...
\begin{tikzpicture}
  % ...
  % drawing altitudes
  \tkzDrawAltitude[draw ...
    =blue] (B,C) (A) \tkzGetPoint{D}
  \tkzDrawAltitude[draw ...
    =blue] (A,C) (B) \tkzGetPoint{E}
  \tkzDrawAltitude[draw ...
    =blue] (B,A) (C) \tkzGetPoint{F}
  % get the orthocenter
  \tkzInterLL (A,D) (B,E) \tkzGetPoint{H}
  % ...
\end{tikzpicture}
```

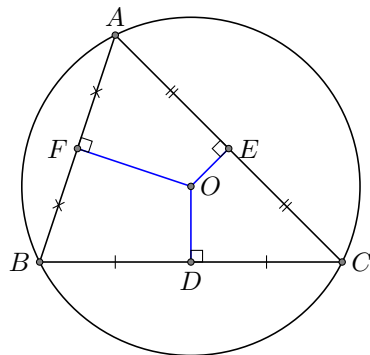


# Circumcircle

```

% triangle_circumcircle.tex
...
\begin{tikzpicture}
  % ...
  % draw the circumcircle
  \tkzDrawCircle[circum](A,B,C)
  % get its center
  \tkzCircumCenter(A,B,C)\tkzGetPoint{O}
  % draw perpendicular bisector lines
  \tkzDrawAltitude[draw ...
    =blue](B,C)(O)\tkzGetPoint{D}
  \tkzDrawAltitude[draw ...
    =blue](A,C)(O)\tkzGetPoint{E}
  \tkzDrawAltitude[draw ...
    =blue](B,A)(O)\tkzGetPoint{F}
  % ...
\end{tikzpicture}

```

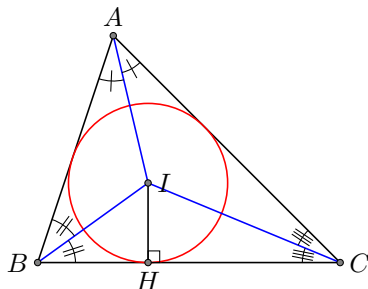


# Inscribed circle

```

%% triangle_inscribedcircle.tex
...
\begin{tikzpicture}
  % ...
  % get the inscribed center
  \tkzInCenter(A,B,C) \tkzGetPoint{I}
  % project it into one edge
  \tkzDrawAltitude(B,C)(I) ...
  \tkzGetPoint{H}
  % draw the circle
  \tkzDrawCircle[draw = red](I,H)
  % ...
\end{tikzpicture}

```



## Euler circle

