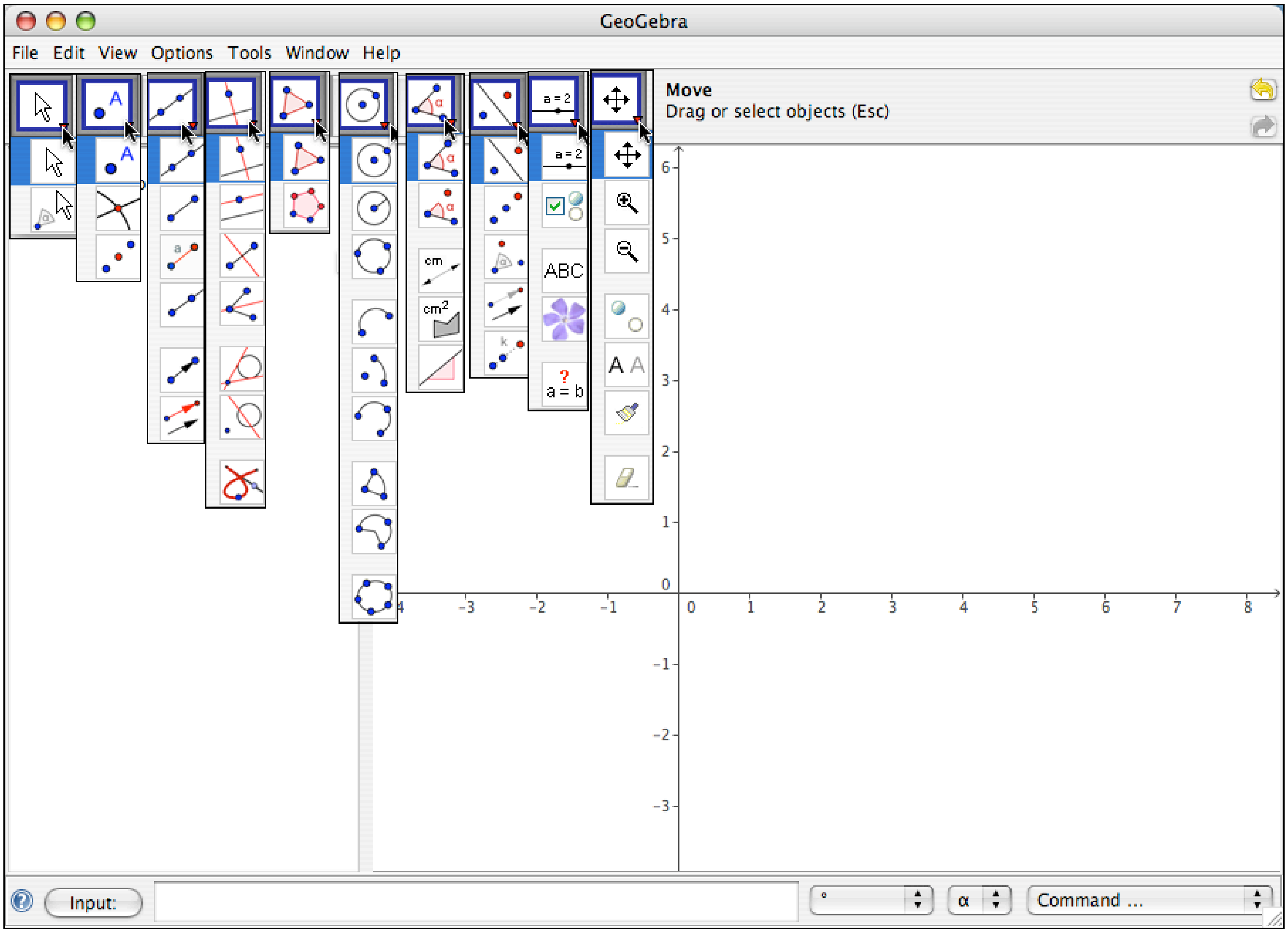
An overview of the functionality of GeoGebra



Many of the geometric object can be created using the icon menus. The composite picture above shows the icons in the pictures. Most are clear enough to understand from the icon. The next page shows each menu along with the text descriptions that accompany the icons.

GeoGebra also has more algebraic commands available from the input line. (The commands invoked by the icons each have text based versions.) I have organized some of those common courses in the pages that follow.

- **Move**
- Rotate around point

- **New Point**
- Intersect two objects
- Midpoint or center

- **Polygon**
- Regular polygon

- Angle
- Angle with given size
- Distance or length
- Area
- Slope

- **Circle with center through point**
- Circle with center and radius
- Circle through three points
- Semicircle through two points
- Circular arc with center through two points
- Circumcircular arc through three points
- Circular sector with center through two points
- Circumcircular sector through three points
- Conic through five points

- **Perpendicular line**
- Parallel line
- Line bisector
- Angular bisector
- Tangents
- Polar or diameter line
- Locus

- **Move drawing pad**
- Zoom in
- Zoom out
- Show / hide object
- Show / hide label
- Copy visual style
- Delete object

- **Line through two points**
- Segment between two points
- Segment with given length from point
- Ray through two points
- Vector between two points
- Vector from point

- **Mirror object at line**
- Mirror object at point
- Rotate object around point by angle
- Translate object by vector
- Dilate object from point by factor

- **Slider**
- Check box to show and hide objects
- Insert text
- Insert image
- Relation between two objects

GeoGebra Commands of interest for conic sections

General

`x(P), y(P)`

Parametric curves

`Curve[expression e1, expression e2, parameter t, number a, number b]:`

Conics - construction

`Conic[point A, point B, point C, point D, point E]`: Conic section through five points *A*, *B*, *C*, *D*, and *C*. Note: No four of the points lie on one line.

`Circle[point M, number r]`: Circle with midpoint *M* and radius *r*

`Circle[point M, segment s]`: Circle with midpoint *M* and radius equal to *Length[s]*

`Circle[point M, point A]`: Circle with midpoint *M* through point *A*

`Circle[point A, point B, point C]`: Circle through three points *A*, *B* and *C*

`Ellipse[point F, point G, number a]`: Ellipse with focal points *F* and *G* and principal axis length *a*. Note: Condition: 2*a* > *Distance[F, G]*

`Ellipse[point F, point G, segment s]`: Ellipse with focal points *F* and *G* where the length of the principal axis equals the length of segment *s* (*a* = *Length[s]*).

`Hyperbola[point F, point G, number a]`: Hyperbola with focal points *F* and *G* and principal axis length *a*. Note: Condition: 0 < 2*a* < *Distance[F, G]*

`Hyperbola[point F, point G, segment s]`: Hyperbola with focal points *F* and *G* where the length of the principal axis equals the length of segment *s* (*a* = *Length[s]*)

`Parabola[point F, line g]`: Parabola with focal point *F* and directrix *g*

Conics – Related information

`Asymptote[hyperbola h]`: Both asymptotes of a hyperbola *h*

`Tangent[point A, conic c]`: (All) tangents through point *A* to conic section *c*

`Tangent[line g, conic c]`: (All) tangents to conic section *c* that are parallel to line *g*

`Directrix[parabola p]`: Directrix of a parabola *p*

`Excentricity[conic c]`: Excentricity of a conic section *c*

`FirstAxis[conic c]`: Principal axis of a conic section *c*

`FirstAxisLength[conic c]`: Length of a conic section *c*'s principal axis

`SecondAxis[conic c]`: Second axis of a conic section *c*

`SecondAxisLength[conic c]`: Length of a conic section *c*'s second axis

`Vertex[conic c]`: (All) vertices of a conic section *c*

`Diameter[line g , conic c]`: Diameter conjugate to line *g* relative to conic section *c*

`Diameter[vector v, conic c]`: Diameter with direction vector *v* relative to conic section *c*

GeoGebra Commands of interest for college algebra/pre-calculus

General

`x(P), y(P)`

Polynomial investigation

`f(x) = x^3 - 3 x^2 + 1`

`R = Root[polynomial f]`: All roots of polynomial *f* (as points)

Parametric curves

`Curve[expression e1, expression e2, parameter t, number a, number b]`:

`Asymptote[hyperbola h]`: Both asymptotes of a hyperbola *h*

`Centroid[polygon poly]`: Centroid of a polygon *poly*

`Circle[point M, number r]`: Circle with midpoint *M* and radius *r*

`Circle[point M, segment s]`: Circle with midpoint *M* and radius equal to *Length[s]*

`Circle[point M, point A]`: Circle with midpoint *M* through point *A*

`Circle[point A, point B, point C]`: Circle through three points *A*, *B* and *C*

`Direction[line g]`: Direction vector of line *g*. Note: A line with equation $ax + by = c$ has the direction vector $(b, - a)$.

`Directrix[parabola p]`: Directrix of a parabola *p*

`Ellipse[point F, point G, number a]`: Ellipse with focal points *F* and *G* and principal axis length *a*. Note: Condition: $2a >$ *Distance[F, G]*

`Ellipse[point F, point G, segment s]`: Ellipse with focal points *F* and *G* where the length of the principal axis equals the length of segment *s* (*a* = *Length[s]*).

`Excentricity[conic c]`: Excentricity of a conic section *c*

`FirstAxis[conic c]`: Principal axis of a conic section *c*

`FirstAxisLength[conic c]`: Length of a conic section *c*'s principal axis

`Hyperbola[point F, point G, number a]`: Hyperbola with focal points *F* and *G* and principal axis length *a*. Note: Condition: $0 < 2a <$ *Distance[F, G]*

`Hyperbola[point F, point G, segment s]`: Hyperbola with focal points *F* and *G* where the length of the principal axis equals the length of segment *s* (*a* = *Length[s]*)

`Div[number a, number b]`: Integer quotient when number *a* is divided by number *b*

`Line[point A, point B]`: Line through two points *A* and *B*

`Line[point A, line g]`: Line through point *A* parallel to line *g*

`Line[point A, vector v]`: Line through point *A* with direction vector *v*

`Mod[number a, number b]`: Remainder when number *a* is divided by number *b*

`Parabola[point F, line g]`: Parabola with focal point *F* and directrix *g*

`SecondAxis[conic c]`: Second axis of a conic section *c*

`SecondAxisLength[conic c]`: Length of a conic section *c*'s second axis

`Vertex[conic c]`: (All) vertices of a conic section *c*

GeoGebra Commands of interest for calculus

General

`x(P), y(P)`

Polynomial investigation

`f(x) = x^3 - 3 x^2 + 1`

`R = Root[polynomial f]`: All roots of polynomial *f* (as points)

`E = Extremum[f]`

`I = InflectionPoint[f]`

`Derivative[f]`

`Derivative[f, 2]`

Standard Calculus

`Derivative[function f]`: Derivative of function *f(x)*

`Derivative[function f, number n]`: n^{th} derivative of function *f(x)*

`Root[function f, number a]`: One root of function *f* with initial value *a* (Newton's method)

`Root[function f, number a, number b]`: One root of function *f* on interval [*a*, *b*] (regula falsi)

`Tangent[number a, function f]`: Tangent to function *f(x)* at *x* = *a*

`Tangent[point A, function f]`: Tangent to function *f(x)* at *x* = *x(A)*

`TaylorPolynomial[function f, number a, number n]`

`Curvature[point A, function f]`: Curvature of function *f* in point *A*

`CurvatureVector[point A, function f]`: Curvature vector of function *f* in point *A*

`OsculatingCircle[point A, function f]`: Osculating circle of function *f* in point *A*

`Integral[function f]`: Indefinite integral for function *f(x)*

`Integral[function f, number a, number b]`: Definite integral of function *f(x)* from number *a* to *b*. Note: This command also draws the area between the function graph of *f* and the *x*-axis.

`Integral[function f, function g, number a, number b]`: Definite integral of the difference of the functions *f(x) - g(x)* from number *a* to number *b*. Note: This command also draws the area between the function graphs of *f* and *g*.

`LowerSum[function f, number a, number b, number n]`: Lower sum of function *f* on the interval [*a*, *b*] with *n* rectangles. Note: This command draws the rectangles of the lower sum too.

`UpperSum[function f, number a, number b, number n]`: Upper sum of function *f* on the interval [*a*, *b*] with *n* rectangles. Note: This command draws the rectangles of the upper sum too.

Parametric curves

`Curve[expression e1, expression e2, parameter t, number a, number b]`:

`Derivative[curve c]`:

`Tangent[point A, curve c]`: Tangent to curve *c* in point *A*

`Curvature[point A, curve c]`: Curvature of curve *c* in point *A*

`CurvatureVector[point A, curve c]`: Curvature vector of curve *c* in point *A*

`OsculatingCircle[point A, curve c]`: Osculating circle of curve *c* in point *A*