# Help for 2-dim Wingeom

Each drawing session with the 2-dimensional program begins with a blank window. One way to fill this window is to go to the **File** menu and retrieve an **Old** figure that has been **Saved**. To create a new figure, however, you will need to add **Points**, **Lines**, **Circles**, or other geometric **Shapes** to the screen, and perhaps apply **Transformations** to them.

In the following guide, boldface type signifies items that can be chosen by clicking with the mouse (menu items and buttons, that is). If a menu item appears gray on the screen, it means that the feature is currently disabled because it is useless. (For instance, you can not delete circles from the figure if there are none.)

The **File** menu:

> A **Saved** figure consists of a single DOS file *filename.ge2* that includes all the construction steps, measurements, highlights, and added text described below. When it is retrieved as an **Old** file, it should appear as it did when you saved it.
>
> Before **Printing** hard copy, you can use **Format** to specify how wide you want the printed image to be. The vertical dimension of the image is then determined by the shape of the window. You can also specify where you want the upper left corner of the image to be. The *x*- and *y-offsets* are measured in inches from the left and top edges of the paper, respectively. You can specify whether you want to have a frame drawn around the image — check the box if you want the frame. If you are not using a *color* printer, it may be best to leave this box unchecked, so that colored lines in the figure do not appear splotchy. These selections are all saved with the figure. Notice, by the way, that the size of text in a printed figure is not determined by the dimensions of the figure — it is set when you choose the font for the text. Thus a small printed image could have disproportionately large text.
>
> **Clip** copies the contents of the drawing window to the clipboard. This bitmap disappears from the clipboard as soon as the *Wingeom* window is closed, however, so if you have plans for the drawing, do them before closing.

Now we turn to the rest of the top menu bar. The three essential elements for building a figure are **Points**, **Lines**, and **Circles**, each one represented by a separate menu.

The **Point** menu provides five ways to add to a figure:

> **Segment Division**. Points on a segment *AB* are described linearly by a coordinate *t*, where *t*=0 means *A*, *t*=1 means *B*, *t*=0.5 means the midpoint of *AB*, and so forth. A *t*-value that is either greater than 1 or less than 0 will extend the segment. Notice that the unit of measure is the defining segment, rather than an absolute scale. If you want to mark a point that is 2 absolute units from *A*, then *t* should be given the value 2/*AB*. (See the remarks on *Numerical Input* below.) Each time that **Mark at** is clicked, a new label appears at the requested location. It is possible to process multiple segments (see *Entering Lists*), as long as all are processed using the same coordinate *t*.

**Circle Division** follows the same rules as does segment division, except that the numerical value runs from 0 to 360, both values assigned to the same reference point on the circle. Random points can also be placed on circles, and dragged along their circles using the mouse. If there are no circles defined, this menu item can not be activated.

It is possible to mark points on segments or circles in *variable* (but precisely known) locations, by using the symbols #, $, or @ in defining the numerical coordinate. These non-alphabetic characters represent numerical values that can be adjusted after the drawing is on the screen. When the value of any parameter is changed (see the **Animate** menu), all points whose positions depend on this value will be adjusted accordingly. Initially, the values of #, $, and @ are 2/3, 1/3, and 1/2, respectively.

Another way to mark new points in the figure is by means of *xy-coordinates*. Click **Grid** to activate this dialog box. When **Mark** is clicked, a new point appears at the place defined by the *x*- and *y*-coordinates on display. Edit these coordinates *before* clicking **Mark**. Points introduced into a figure in this way can not be dragged. If you type a formula into the box, it is parsed. For example, setting *x* equal to *[sqr](3)* is understood to mean *x* = 1.732..., and setting *y* equal to *[pi]* is understood to mean *y* = 3.14159... Like the two preceding dialog boxes, this box can stay on the screen indefinitely. Because the input boxes only allow for 20 characters, long formulas can be typed in only if they have been prepared in advance. See **Edit/Functions** below.

Although the axes do not show unless they have been requested, the underlying system for all *Wingeom* figures is coordinate geometry. All data is maintained and all calculations are carried out with 18-digit precision in the background. Coordinate information about any figure is readily available. (See **Buttons** below.)

The fourth source of new points is **Intersections**. There are three types: **Line-line**, **Circle-circle**, and Line-Circle, called **Mixed** in the menu. For all three, clicking **Mark** causes a new label to appear, at the intersection defined by the boxes in the dialog box. In the two circle cases, there may be two intersections, and both will receive labels. If the **Other/Autoextend** item is checked (see below), segments are extended to show intersections. Unwanted segments or points can be deleted. Intersection points that are already labeled will not receive new labels.

The **right button** can also be used to mark new points – see **Buttons** below.

The **Line** menu provides several ways of adding to a figure:

One way is to join two points by a segment. The can be done by **connecting** with the left button, or with the **Segment** dialog box, in which you can request multiple segments (see *Entering Lists* below). It is also possible to concatenate segments – for example, type *ABCDA* to draw quadrilateral *ABCD*. Or enter *PQRP/A'M* and the program will draw the segments *PQ*, *QR*, *RP*, and *A'M*. All characters other than the vertex characters (commas, for example) are read as separators.

If you request a **Line** through two points, the program draws the segment and extends it to the screen boundary in both directions.

You can switch segment **Extensions** on and off. There is a difference between extending *AB* and *BA*, of course. Doing both produces the *line* through *A* and *B*. If you extend a segment whose extension is already on the screen, the extension will be turned off. It is occasionally desirable to place a point on the extension. The program will do this for you if you add a third letter – it does not matter which – to your request. In other words, entering *ABX* will extend *AB* and place a random point on the extension (using the first available label).

The **Bisect old angle** dialog box allows you to enter multiple requests for *angle bisection*. Each angle must be specified by a *three-letter designation*, the vertex labeled by the middle letter. Each bisector appears as a ray, with a new point on it.

The **New angles** dialog box requires that you *describe* the angle you want. The main complication is that you must be careful of angle *sense*. The program distinguishes between angle sizes of 60 and –60, for example. You can copy another angle, by entering its name (use < to signify angle) as the angle size. Here you must also be careful not to confuse <*ABC* and <*CBA*, for example.

In the **Parallels** dialog box, a segment and a point are needed to define the new line. Press **Mark** to draw the line, which will have a new point on it. If the new point is off the screen, the program will adjust the view frame.

There are three dialog boxes for perpendiculars. The **Altitudes** and **General perpendiculars** dialog boxes are just like the parallels dialog. In the former case, only a segment is drawn – the implied intersection point is automatically labeled. The **Perpendicular bisectors** dialog allows you to input a list of segments, each of which will have its midpoint automatically marked and the bisector drawn.

There are four ways to add **Circles** to a figure:

The simplest is to request the circle that goes through three non-collinear points. This is the **Circumcircle**. When the circle appears, so does a label for its center. This dialog box can process multiple requests.

You can also request the **Incircle** of a triangle. The circle appears along with labels for its center and the three points where it is tangent to the triangle.

The **Center-radius** dialog box allows you to define a circular arc by means of its center, its angular size, and its radius. If the radius is given in the form *AB*, where *A* is the center and *B* is a point already on the screen, then the arc will begin at *B*; otherwise, the program will choose a random starting point (at the correct distance from the center). If the arc is not a full circle, then the program will label the other end of the arc. If the arc size is defined as <*BAC* and the radius as *AB*, the program will try to assume that *B* and *C* are supposed to be the ends of the resulting arc, provided that *AB* and *AC* are equal (to within a small tolerance).

The left button can be used to draw circles — see **Buttons/Circles** below

If circles look too polygonal, you can increase the **Density** of the plotted points, by entering a positive integer (the default is 1). Increasing the density makes the circles look smoother, but it also increases the drawing time.

The available **Shapes**:

**Triangles** defined by **ASA**, **SAS**, or **SSS** data.

**Polygons** (either **regular**, having a specified number of sides, or SAS-defined **kites** and **parallelograms**). The regular polygons can be **attached** to existing segments. The *Sides* entry describes the figure to be attached. Because the program prefers to label polygons in a counterclockwise sense, there is a difference between attaching a polygon to segment *AB* and attaching a polygon to segment *BA*. The polygon attached to *AB* is to the left of *AB*, as you look from *A* towards *B*. After listing the edges to receive polygon attachments, press **OK** to see the results. You may concatenate the list – to attach squares to the outside of pentagon *ABCDE*, enter the edge list as *AEDCBA*, rather than as *AE,ED,DC,CB,BA*.

**Random** circles and polygons (namely **triangles**, **right triangles**, **parallelograms**, **kites**, **trapezoids**, **convex** or **cyclic** polygons with a prescribed number of sides).

**Segments** of specified length.

A **Duplicate** of a subfigure, defined by the vertices that appear in it. As in *DEFGM"*, the list need not have commas.

A **conic section** defined by three points. If an **Ellipse** or a **Hyperbola** is desired, the first two points are the focal points, and the third is on the curve. If a **Parabola** is desired, the first two points define the directrix and the third is the focus. Conic sections can be **Removed** at any time.

These shapes appear at random places near the center of the screen. They can be dragged around the screen, independently of the other parts of the figure.

The **Transformation** menu:

One way of adding to a figure is to apply a geometric transformation to some of its elements, thereby creating new elements. For example, a figure can be augmented by reflecting each of its points and each of its segments in a specified line. This is what you are allowed to do in the **Mirror** dialog box. The **Mirror** need not be an actual line in the diagram, but the points used to define it must exist. In the *Apply To* edit box, list the points whose images are to be calculated. The only new segments and circles that will automatically appear are images of segments and circles connecting the designated points.

Similar remarks apply to the **Translate**, **Rotate**, **Dilate**, and **Glide-reflect** menu items. In each case, you first define a transformation, specify the points to be transformed, then click **OK** to complete the task.

The new points are labeled by adding primes to the old labels, if possible. If the **Label Saver** item is checked, the program will assume that image points that are very close to existing points should actually coincide. You can override the program choices by re-labeling (see **Buttons** above).

To automatically apply the most recent transformation to the most recent set of images, press *F7*, or click **Again**.

There are various ways to **Edit** your figure:

The most recent construction can always be **Undone**. Repeated use of this option will eventually remove everything.

Points, segments, and circles can be **deleted**, subject to the following:

A point can not be deleted if it is needed to define an existing circle.

A segment can not be deleted if there is a point marked interior to it.

Deleting a point also deletes every segment that *ends* at that point.

The most recent construction can always be **Undone** (*Alt+Bksp*). Repeated use of this option will remove everything. If you change your mind immediately, you can regain undone constructions by **Undo Undo** (*Ctrl+Bksp*).

If you elect to **Delete all**, you will be left with a blank screen again, of course.

The mouse can be used to drag vertices around the screen, as described below under **Buttons**. The **Grid** dialog box allows you to slide vertices to *precise* locations, defined by coordinates. Click the **Move** button when the coordinates and the vertex label have been properly entered. If the vertex is movable, the result should show at once.

New points are assigned labels automatically by the program. Like any text on the screen, they can be edited and moved. (See **Buttons** below.)

Click **Decimals** to alter the number of places shown (in Measurements, for example.).

Any figure can be used to define a **Macro**. This means designating a set of defining **Variables** (vertex labels) and a sequence of steps between a **Starting** step and a **Stopping** step. These can be defined (here) as soon as the figure is created, or later, when it is recalled into a macro window (see description below).

The **Header** item refers to an optional 60-character description that introduces the History list.

To type long formulas into input boxes that only allow 20 characters requires defining and naming **Functions** ahead of time. The input box for this dialog will accommodate an 8-character name and a 50-character formula, which must be written using the dummy variable *x*. References to your function library must be enclosed in square brackets, just as with the standard functions described below.

Another way of editing your figure is to **Randomize** it, which affects only those figures that contain a random Shape. The program redraws the figure, selecting different random shapes (of the same kind).

The **Measurement** dialog box:

Here you can make almost any request for numerical data – as long as it can be typed into the edit box, which accepts up to 60 characters. The conventions are illustrated by the following examples, which can be entered in lower case:

Type *AB* and press *Enter* to see the length of segment *AB*.

Type <*ABC* and press *Enter* to see the size (in degrees) of angle *ABC*.

Type *ABC* and press *Enter* to see the area of triangle *ABC*.

Type *ABCDE* and press *Enter* to see the area of convex pentagon *ABCDE*. For a non-convex figure, the displayed answer *might* still be correct.

Type *AB+BC+CD+DA* and press *Enter* to see the perimeter of quadrilateral *ABCD*.

Type *AB/AC* and *Enter* to see the ratio of the lengths of segments *AB* and *AC*.

Type (*AB*^2+*BC*^2)^0.5 and press *Enter* to see the length of the hypotenuse of the right triangle with legs *AB* and *BC*. Or, press *F2* to place $\sqrt{(}$ in the input window, then type *AB*^2+*BC*^2) to finish the description. The exponents can also be inserted with *F3*.

Type *A* and press *Enter* to see the coordinates of *A*.

Type *[sin](<ABC)* and press *Enter* to see the sine of angle *ABC*.

The **Measurement** dialog box can not be left on the screen while drawing is taking place, so each new calculation is placed in the drawing window, where it will change dynamically with the figure. If you do not want a particular item to appear in the drawing, select it (click it) and then click **Hide**. Click **Show** to reverse this process. (Displayed measurements can be moved around like other text items, by the way; just put the mouse in **Btns/Text** mode.) To permanently remove a measurement from the master list, select it and click the **Delete** button. To recover this measurement, you will need to retype it. Multiple selections are possible. To insert new measurements into the list, the cursor must be blinking in the edit box. When an item is first entered, it is automatically selected. To copy an expression back into the input box, select it and then click the **Copy** button. As shown above, the program will understand requests for standard functions, provided that the function names are placed inside square brackets. To insert $\pi$ = 3.14159... into the edit box, you can either type [*pi*] or press *F1*. Press *F2* for a square root sign, *F3* for the exponent $^2$, *F4* to ask for the length of a circular arc (if the center is *B* and the arc endpoints are *A* and *C*, then $\Omega ABC$ gets the desired length), *F5* for the golden ratio, and *F6* to ask for the perimeter of a polygon, as in $\sum(ABCDE)$. The program also recognizes the functions

*sin, cos, tan, csc, sec, cot, arcsin, arccos, arctan, sqr, log, ln, exp, int, frac, sgn, pi*.

As the example above shows, it is necessary to enclose the function name in square brackets, so that the program does not try to read the letters as the name of a polygon.

You can also request the slope of a segment AB by entering *[slope](A,B)*, and its slope-intercept equation by entering *[eqn](A,B)*.

Because the *x*- and *y*-input boxes only allow 20 characters, it is necessary to prepare long formulas in advance. (See Edit/Functions above.)

In the **Buttons** menu, the functions of the left and right mouse buttons are indicated by check marks:

> In **Segment** mode (the primary drawing mode), the right button creates new points in the diagram. Point anywhere and click the right button. A new vertex will appear (unless the click was too close to an existing vertex). If the click is close to a segment or an arc, the point is placed *on* that segment or arc. If the click is close to the intersection of two segments (arcs), the new point is the *intersection* of those segments (arcs). If the click is close to the concurrence of three things, the intersection of only two of them will be labeled, however. The left button creates new segments by connecting vertices: Point at any vertex, hold down the button, slide the mouse pointer to another vertex, and release the button. A new segment should appear (unless the points were already connected, or unless the pointer was not accurately positioned).
>
> In a similar fashion, **Circles** can be created. Point at any vertex, hold down the button, side the mouse pointer to another vertex, and release. If the mouse is not released near another vertex, the program will label a new point on the circle.
>
> Another use of the left button is to reposition (*drag*) vertices: To put the mouse into this mode, click **Btns/Drag points**. Then point at a vertex and press the left button (but do not release it) — the target vertex should change color to signify that it has been engaged. Slide the mouse and the vertex should move. What happens to the rest of the figure depends on the vertex. If its definition depends in any way on other parts of the figure, then the whole figure will slide. For example, you can not slide the constructed midpoint of a segment without sliding the whole segment (and whatever that is attached to, etc). On the other hand, certain vertices and shapes are independent of the rest of the figure, and so can be slid around, while the rest of the figure reacts accordingly (or not at all). For example, the vertices of a random convex quadrilateral can be slid independently of each other. Subsequently defined points (the midpoints of the sides, say) would then move as required by their definitions. If a point has been placed randomly on a segment or an arc (which can be done using the right button — see above), you can slide the point using the left mouse button. (If a point has been placed on a segment or arc using a variable coordinate, you must use the appropriate **Animate/slide** dialog to move that point.)

When the left button is in drag-pt mode, the right button is used to change the label type of selected points. Right-click a vertex again and again. You will see its label change from alphabet to nothing to a filled circle to an open circle, etc.

Even though it might not show, the underlying surface for all figures is graph paper, and all points are maintained using *xy*-coordinates. These coordinates are accessible with the left mouse button, once it has been reassigned — click the **Coords** entry in the **Buttons** menu. Then point at a vertex and press the left button. While it is held down, its coordinates are on display. Notice that sliding the mouse has no effect. The accompanying function for the right button is to zoom. Point at any vertex of the figure and click the right button — you should see a two-fold magnification of this part of the figure, with the click point now at the center of the screen. The 2X zoom power can be changed by selecting **Zoom Factor** in the **Other** menu. Notice that changing the factor to a value less than 1.00 would have the effect of making a zoom-out button, which is not really needed. Notice that clicking the right button on an empty part of the screen, or using too high a magnification factor can produce an empty screen. Then you will have to use View/Window to restore the figure to normal appearance (see below).

Another use of the left button is to reposition text, labels, or measurements. To activate this feature, click **Btns/Text**. Then point at any inserted text, vertex label, or measurement in the figure, hold down the left button, and drag the text to its desired position. The accompanying function of the right button is to request an *insertion* of text into the figure. Right-click any point in the drawing window. A dialog box opens, which has an edit box for up to 60 characters of text, and buttons to select the color and font. Type the desired text into the edit box. When you close the box by clicking **OK**, the text appears. The text is placed with its upper left corner at the pointer. If you point at existing text, this text is reopened for editing. Point at a vertex label if you want to re-label the vertex. The program will not let you use the same label twice.

If the figure is altered, or the window resized, the vertex labels move with their vertices. The positions of other text items are defined with respect to the screen center.

If you find it convenient to have a toolbar for changing the function of the mouse buttons, click **Toolbar**. The resulting dialog box can be left on the screen wherever you want it.

Click **Re-center** to move all the alphabetic labels back to their vertices.

Click **Home** to bring all measurements back to the upper left corner of the screen.

The **View** menu:

To bring all parts of a figure into view, click **Window**. (or press *Ctrl+W*.) This automatic framing of the figure can be fine-tuned by either dragging the figure with the mouse (click a secondary vertex) or else by using the **Center-Width** dialog box, which displays the coordinates of the screen *center* and the *width* of the frame. Replacing this width by a smaller value makes the figure appear larger, fitting more tightly in the frame.

To radially enlarge the viewframe, click **Expand** (this makes the drawing seem smaller). To make the drawing larger, **Shrink** the frame. The keyboard combinations *Ctrl+E* and *Ctrl+S* accomplish the same effect without the mouse. The expansion ratio is set by clicking **Factor**. If this value is made smaller than 1.0 (the default is 1.05), the effects of the preceding buttons are reversed.
It is occasionally desirable to **Rotate** the figure within its frame (*Ctrl+R* from the keyboard). Because the underlying (invisible) graph paper always has its axes parallel to the sides of the frame, this means that *the coordinates of the points will be altered*. The **Angle** dialog box allows you to specify the *angle*; the rotation center is the center of the screen (whether or not there is a vertex there). A positive angle means that the figure will turn counterclockwise.

**NB**: This *relocation* of points is incompatible with certain constructive steps (**Point/Grid**, for example), and the program might therefore get confused — especially when you ask the figure to regenerate itself (by dragging its vertices, for example).

The underlying coordinate axes are generally invisible. To keep them on the screen, click **Grid** in the pull-down **View** menu.

When the **Units** item is checked, the program appends descriptive labels to most measurements. Uncheck this item if you do not want to see them.

It is occasionally desirable to hide some or all of the labels, or to mark the vertices with bullets or circles. This can be done by using the **Labels/Individual** dialog. The default is to label all vertices alphabetically. Notice that consecutive labels can be abbreviated by a hyphen; thus K–P means K,L,M,N,O,P. Each click of the **Mark** button changes the status of a single group of vertices, according to the state of the **Label** checkbox and the radio buttons **bullet, circle,** and **nothing**.

Click **Label/Font** to alter the font used to display the labels. When you close this dialog box by clicking **OK**, the figure is relabelled with the new font.

Click **Label/Hide/show** (or press *Ctrl+L*) to hide (or show) all the labels.

Click **Offset** to move all labels away from their vertices.

To bring all vertex labels back to their default positions (centered on the points) click **Center**.

Click **Dot** mode (or press *Ctrl+D*) to change the icon used to mark the vertices (nothing, filled circle, open circle). If the labels are centered on the vertices, you will not see any change.

To change the alphabetic labels, put the mouse into **Buttons/Text** mode and right-click the labels that need altering.

You can **Swap** the labels for two vertices. This is useful when the program has permanently assigned a desired letter to a hidden vertex (see below).

The text and labels can be displayed in a variety of **Fonts** and sizes (whatever your installation of *Windows* has on hand).

There are several ways of *highlighting* a figure:

Individual lines and circles can be drawn in contrasting colors – click the **Highlights/Color line&circle** menu item. The resulting color-change dialog box allows you to re-color individual elements of the figure. Each click of the **apply** button changes the color of one (or all) of the lines, or the color of one (or all) of the circles. In particular, you can hide lines and circles (without deleting them) by re-coloring them *invisible*. The entry can define either a segment or a center-point ray, depending on the state of the radio buttons. An entry is overridden by clicking the **All** button. The dialog box must be closed to return to drawing.

Just as lines and circles can be re-colored, they can also be drawn in different styles and widths. Use the **Highlights/Style** dialog to **apply** a new style to one (or all) of the lines (or circles) in the figure. Notice – as in re-coloring – that the requested change applies to the whole line (circle) that is defined. (If you want to proceed on a segment-by-segment basis, it is necessary to construct the figure so that the program does not recognize segments as collinear.) Adjusting widths of lines is useful for highlighting them on a black-and-white printer. The checkbox in the dialog allows you to fill a circle or sector with the color used to draw the arc.

Polygons and circles (or sectors) can be highlighted by filling their interiors with colored patterns. The **View/Highlights/Fill** dialog box allows you to select *Color* and fill *Pattern*. Type the name of the polygon in the edit box, then click **Fill**. The list box shows the sequence of filled regions. Because each region could be covered by any that follow it in the list, you may find that you want to edit the list, by deleting selected items and re-inserting them. If the radio button *First* is selected, then each new **Fill** goes to the top of the list; otherwise, it follows a highlighted entry — if there is none, it goes last in the list.

To indicate segments that are parallel (or have the same length), or angles of the same size, it may be useful to place arrows (or ticks) on the segments, or to mark the angles with small arcs or brackets. The sizes of all markings can be adjusted, by clicking **Tick**, **Bracket**, **Arrow**, or **Arc**. Click the **View/Highlights/Markings** menu item. The resulting dialog box allows you to select the type of marking and the number to be placed at the location specified in the *Where* box. Click **Add** to make the mark, or **Remove** to delete it. Click **Remove All** to clear the figure of markings. Changes are made immediately.

The **Animation** menu:

One way to dynamically alter a diagram is to define it in terms of parameters; the symbols #, $, and @ stand for global values (initially 2/3, 1/3,and 1/2, respectively) that can be changed at any time. Any figure that has been made to depend on either of these values is immediately altered as a result. The following remarks apply equally well to all three parameters. For example, if an angle is drawn with size 90#, it will open and close as the value of # is altered. Click **Animate/# slide** to control the value of #. The displayed value can be altered by sliding the scroll bar, or by typing a desired value into the box and pressing *Enter*. Click **autoreverse** to make the program slide the scroll bar *back and forth* for you. Click **autocycle** to make the parameter values *jump* from one end of the range back to the other. Press *F* or *S* to change speeds, and the spacebar to change direction. Any other keypress stops this animation. While the program is in this autopilot mode, the dialog box disappears. The range of the scroll bar is initially from 0.0 to 1.0, but it can be reset. Clicking the **set left** button sets the leftmost scroll position to the displayed value, and clicking **set right** sets the rightmost scroll position to the displayed value. Doing both inadvertently could reduce the range to a single value, so the program will ignore such a request. You should type the desired range values into the box *before* pressing **set**. If you enter a value into the box that lies outside the existing range, the range is reset (upward or downward) automatically.

Click **Tracing** to open an inventory dialog box that lists all the graphs. Then click **new** or **edit** to define a tracing. In addition to the color, you must make other choices:

The tracing can be controlled by one of the parameters #, $, and @. A figure that has been constructed using # to define its elements will change when the value of # is changed; in particular, any designated vertex can be expected to trace some sort of locus as # runs through a specified range of values. To choose one of the three parameters, click the corresponding radio button.

The tracing can also be controlled by one of the randomly marked vertices that slide along segments or circles (see description above). If this is the case, click the radio button and type the label into the box.

In either case, the controlling element is confined to the range of values that appears in the boxes marked *Low* and *High*, and it is incremented as many times as *Steps* indicates. If a parameter controls the tracing, the low-high data defaults to the range that currently limits the parameter's scroll bar.

The *pen* is placed on the drawing vertex, if the corresponding radio button is marked. You can also choose to draw the *Whole* figure (instead of just a single vertex) for each of the designated control values. If the whole figure option is selected, it is possible to apply the *labels* (either letters or circles at each designated vertex) to each drawing.

When you close the box by clicking **OK**, the new tracing takes place, then the figure is superimposed on the graph. If the window is re-sized, re-centered, or re-scaled, the process is repeated for all the current tracings.

If there is already tracing in the background, this will be left as is. In the **Tracing** inventory, click **edit** to alter an old tracing, or **delete** to remove it. Click **delete all** to remove all tracings. Any such change requires that all remaining tracings be redrawn.

When vertices are dragged around the screen, the tracings will not be automatically updated, unless the **Monitor** item is checked. The extra drawing will make the mouse seem unresponsive, however, so it is better to avoid this item and just click **Retrace** (or press *Ctrl+X*) every now and then.

**Temporary tracing** is enabled by clicking this item and entering a list of vertices that are to leave trails whenever the figure is altered. This list is emptied whenever a permanent tracing is made.

Click **Graph** to open a secondary drawing window, in which the value of one measurement can be plotted versus another. This window is not available unless at least two measurements have been entered. The resulting window has only a small number of items on its menu bar. One addition to the **View** menu is the option to define the window by specifying the coordinates of its corner points. (This overrides the program directive to use the same scale for both axes.) The **Graph** menu allows you to specify the two measurements that the axes represent...click **Variables**. Click **Draw** to make the graph. This dialog box is similar to the **Tracing/new** dialog described above, in that the graph must be controlled by a parameter, or by a point constrained to move along a line or circle. Click **Erase** to remove the previous graph. Click **Home labels** to move the axis labels back to their default positions. They can be dragged around the screen by using the left mouse button.

The **Other** menu contains miscellaneous features:

The **History** list shows step-by-step how the figure on the screen was created. It begins with an optional description of the figure, and ends with the current values of the parameters #, $, and @, and a list of any user-defined functions. This text file can be **Printed**, **Saved**, or put on the **Clipboard**.

The **Segment** list does just what the name implies. It is useful because it is not always obvious which points are regarded as collinear by the program, which makes no assumptions about coincidence or concurrence. Unless the program is told explicitly that a particular point falls on a particular line – via intersection or segment division, for example – it does not assume such facts. Nor does the program assume that points that are close together are meant to be the same point.

The *data* is an accumulation of numerical measurements (up to five) on display in the drawing window. (Unless there are such displayed items, there can be no data.) To collect data, the **Accumulate Data** item has to be enabled (checked). Then the list grows by one row each time the drawing changes dynamically (via dragging vertices or varying the parameters). Data collection is signaled by a color change in the displayed data. To see the data, click **Inspect**. This text file can be **Printed**, **Saved**, or put on the **Clipboard**. Click **Delete** to erase the list and start again.

The program views segments as parts of lines. In **Autoextend** mode, you can slide points past the ends of segments, and non-intersecting segments will be extended if necessary to show a labeled intersection point. Click this menu item to change the mode.

It is possible to retain the displayed measurements in a **Data** array. Check **Collect** to enable this process. When this item is checked, the data array grows by one row whenever the value of # (or $ or @) is changed. The displayed values change color to indicate that the collection process is enabled. To see the current array, click **Inspect**. To empty the array, click **Zero** (or **Delete** in the **Inspect/File** menu). To save the data as a numerical array (so that *WinStat* can read it), select **Save**.

The **PicTeX** item converts the current figure into a text file that can be imported into a TeX document. Enter the width of the desired figure and the box number that it will have in the document. The vertical dimension is determined by the shape of the window.
Open a **Macro window** to apply previous constructions without having to redo them. This window is equipped just like any other 2-dimensional window, except for its **Other** menu. There you can define a macro by designating a block of lines (in its history). You simply specify the **Start** and **Stop** lines (select each separately). You also need to specify the list of vertices that will be mapped to a corresponding list in the parent window. Click **Macro vbls** to do this. Make sure that all the vertices that appear in the start-stop section of the history are indeed derived from the defining **Macro variables**, otherwise the program will not know what to do (and may do something strange instead). Each time you **Apply** the macro, you will be asked for the corresponding list. When you save a drawing using the Macro window's **File/Save** menu, the macro choices are saved with it.

As many as *four* macro windows can be open at a time for a given parent window.

Click **Replay in Slow Motion** to step through (using the spacebar) the construction of the current figure.

Click **Scrollbar/Calibrate** to change the number of scrollbar steps. The default is 100; it can be any integer from 2 to 32767.

The **Home** button moves all the displayed measurements to the upper left corner of the drawing window.

The **3D Solid** dialog box allows you to create drawings of 3-dimensional figures, specifically prisms and pyramids. To define such a figure, you choose a polygon to serve as a *base*, by listing its vertices. Then you define the upper base (for a prism) or the vertex (for a pyramid) by specifying the three components of a vector *offset*. Click **OK** to see the figure, which appears in its own 3-dim drawing window.

On-screen **Help** consists essentially of this document. Click **Find** and **Next** to search for keywords. The search is not case-sensitive, and it always begins at the cursor position.

**General remarks about 2-dim *Wingeom***:

*Labels* are assigned sequentially by the program. They can be changed (see **Buttons** above). Initially applied directly over the points they represent, they can also be moved. There are 234 = 26*9 possible vertex labels, namely *A*, *A'*, *A"*, *A%*, *A&*, *A?*, *A~*, *A`*, *A!*, *B*, *B'*, and so on. The program makes no assumptions about the points it labels, and may occasionally assign what *seems* to be a superfluous label to a point that is already labeled. Actually, the program is labeling what it thinks is a different (nearby) point.

For example: Three lines are concurrent at a point, which can be labeled by asking for the intersection of any *two* of the lines. This means that there are *three* different ways of labeling the point of concurrence. To the computer, the three possible labels represent three *different* points (they may actually differ from one another in the 18th decimal place). In other words, the computer does not assume (or know) that the intersection of two of the lines is actually on the third line.

This principle sometimes can be confusing – when you ask the program to delete a segment and nothing happens, for example, it may be because you think that a certain point is on a certain line, but the program does not think so.

*Circles* are defined in terms of two points – the center and a point on the circle. For this reason, it is not possible to delete the central point, nor a solitary defining point on the circle, if the circle remains active. As points are added to a circle (via intersection, say), other non-central points become available for deletion.

*Cases:* There is no distinction between upper and lower cases. Entering segment *ab* has the same effect as entering *AB*.

*Angles*: Identification requires *three* letters, even in situations where one would suffice. When describing angles, positive values signify counterclockwise turning. For example, the descriptions 100 and –260 are equivalent. In the **Measurement** window, however, angle sense is *not* significant; there is no difference between <*ABC* and <*CBA*.
*Entering Lists*: Use commas or spaces or / to separate segments, points, triangles, etc.

*Numerical Input*: In addition to decimal values, the program understands expressions such as *AB*+*BC* and *AC/BC*, and so on. The size of a new angle can be entered as <*ABC*, which causes the program to copy angle *ABC*, in both sense *and* size. A new point can be placed on a segment *AB* by providing the coordinate *PC/PD*, which causes the new point to be placed in the *same* relative position on segment *AB* that *P* occupies on segment *CD*. There are also three constant parameters, represented by #, $, and @, that can be used in formulas, and redefined at will. For instance, to mark a variable point on a segment *AB*, use # as the coordinate. A point defined in this way can be moved (along its segment) by changing the value of #. *This enables different parts of the figure to be dynamically linked*. See the **Edit** menu.

When retrieving figures that have random elements, do not be surprised if the figure looks different, for the retrieval process re-randomizes all random shapes.

Many dialog boxes (segment division, for example) are permitted to stay on the screen indefinitely. Others (measurement, for example) must be closed when you want to begin another activity. Any active dialog box can be closed by clicking the system button in the upper left corner, by pressing Alt-F4, or by pressing *Escape*.

When a window or dialog box is active, its title bar is highlighted. To select different parts of an active dialog box, you can use the Tab key instead of the mouse.

**A quick tour of the program**

Start *Wingeom* by double-clicking its icon, or by clicking the icon once and pressing *Enter*. Click the **2-Dim** item on the main menu bar. This will create a small drawing window. From now on, we use its menu bar. Menu items in boldface are meant to be clicked with the left mouse button.

With the mouse in **Btns/Segment** mode, right-click three separate places in the drawing window. This should create three random points, labeled *A*, *B*, and *C*. Now put the pointer on *A*, hold down the left button, drag the pointer to label *B*, and release. This draws out segment *AB*, which should persist on the screen after you release the mouse button. If not, try again. In the same way, draw out segment *BC*. Angle *ABC* should now be visible. Finally, right-click a point that is on segment *AB* (or at least looks that way). The resulting label *D* should appear on the segment. You will test this point shortly.

You have just seen how the mouse can be used to mark new points in the diagram, as well as to connect them to each other by segments. To make the mouse perform other useful drawing services, its mode of operation must be changed. Put the mouse into **Btns/Drag points** mode by clicking this item.

Point at vertex *B*, hold down the *left* button, and drag the mouse. Instead of seeing a developing segment attached to *B*, you should see vertex *B* move across the screen. It stops as soon as you release the button. While engaged by the mouse, the color of the label changes, and the mouse pointer icon disappears. Now *right*-click vertex *B*, without moving the mouse. The label should disappear. Do it again (right-click the vertex where the label used to be). The vertex is highlighted by a filled circle. Do it again. The vertex mark changes to an open circle. Do it one more time and the original label *B* reappears.

Click **Measure**, which opens a dialog box where measurements can be requested. For example, you find the size of angle *ABC* by simply typing <*ABC* into the edit box (where the cursor should be blinking) and pressing *Enter*. Notice the use of the less-than sign to indicate an angle. The angle's size (in degrees) shows in the dialog's list box, and it is also written in the top left corner of the drawing window. The cursor goes back to blinking in the edit box, ready for the next request. Type *AB* and press *Enter*. The length of segment *AB* appears (in two places). Another paragraph explains what this length means.

In order to return to the diagram, the **Measurement** dialog *must* be closed, which can be done using the box in the corner, or (more simply) by just pressing *Escape*.

Click **View/Grid**. You will notice that coordinate axes are added to the picture. This shows that all operations are actually taking place on a sheet of graph paper that is often invisible. The displayed length of *AB* refers to this absolute coordinate system. If you want the graph paper invisible, click **View/Grid** again (this item acts like a toggle switch).

The mouse should still be in **Drag points** mode, so use the left button to drag any of the three vertices around the screen, and notice that the displayed measurements change accordingly. In particular, notice what happens if you try to drag vertex *D*. You should find that it only slides *along* the line *AB*, where you placed it. (If you can drag *D off* the line, then the click that placed *D* there was not close enough to the line after all.)

Here is an experiment: Drag a vertex *out of the drawing area* and release the button! This is a deliberate attempt to leave the vertex in an inaccessible spot, as if to lose it forever. To restore the complete diagram to visibility, however, click **View/Window**. You can also just press *Ctrl+W* (two keys simultaneously), a keyboard shortcut that is listed in the **View** menu. The viewframe should enlarge and re-center itself so that everything shows again. If the grid is still showing, you will notice that the coordinates of the points have not changed.

Click **Circle/Radius-center**. The resulting dialog box allows you to add circles to the diagram by specifying their central points and their sizes (radii). Type *B* into the *center* box and *BA* into the *radius* box. Then click **Mark** (or just press *Enter*). A circle should appear, going through *A*, its center at *B*. (Incidentally, the *radius* box will interpret a variety of requests, in addition to this self-explanatory example.)

Put the mouse back into **Btns/New point** mode, and right-click the point where the circle intersects segment *BC*. This point should acquire a label (probably *E*) and the other point where the circle meets the same line is labeled, too.

Put the mouse back into **Btns/Drag point** mode, and watch what happens when you use the left button to try to drag one of these newly labeled intersection points. The whole diagram slides (graph paper, measurements, and all). This is because there is nothing else that the diagram can do, without ruining its integrity. Other points in the diagram (*A*, *B*, and *C*) move independently, forcing child points (including the circle and the segments) to react. Point *D* is capable of limited movement (because of its definition), but *E* and *F* are completely determined by their parents. This is actually a good way of fine-tuning the placement of a diagram within a window – just slide an immovable point.

Measurements automatically appear in the upper left corner of the window, but they can be moved around, both inadvertently (as in the preceding paragraph) or purposely (see next paragraph). One way to find missing measurements and reposition them in the corner of the window is to click **Other/Home measurements**.

Put the mouse into **Btns/Text** mode. The left button can now be used to drag any text item in the drawing window to a new location. This includes measurements, alphabetic vertex labels, and supplementary text, which are introduced (new) by using the right button: Point at a spot where you want text to appear and click the right button. An input dialog box appears, into which you can type the text. You can also choose the display font, its size, and its color. When you close this box by clicking **OK**, the text appears. If you right-click existing text, it is reopened for editing. If you right-click a vertex label, you get a different dialog box, which allows you to re-label the point. To change the font or the size of the vertex labels, click **View/Labels/Font**. The resulting Windows font dialog box lets you experiment with whatever fonts your installation has available.

The point *D* was placed randomly on segment *AB* using the mouse. It can be slid freely along the line using the mouse, but the precise position of *D* is uncontrollable. There is another way of marking points on segments that does allow total control. Click **Point/Segment division**. The resulting dialog box prompts you for two things: A segment name and a coordinate. Type *BE* into the *segment* box, and notice that 0.5 already appears in the *coordinate* box. Click **Mark at**. You should see a label appear at the midpoint of segment *BE*. That is what the 0.5 means, for 0.0 signifies *B* and 1.0 signifies *E*. Now for something *new*: Type *BF* into the *segment* box, type # into the *coordinate* box, and click **Mark at**. A new label appears on segment *BF*, roughly two thirds of the way from *B* to *F*. Press *Escape* to close this dialog box.

The character # is interpreted by the program as a *variable* number. Its actual value when the drawing window opens is 2/3, which is why the new point looks like it is two thirds of the way from *B* to *F*. To appreciate the versatility of this construction, click **Animate/# slide**. This creates a dialog box that displays the current value of # (it should be 0.666...), as well as a variety of ways of *changing* that value. One way is to slide the scroll bar using the left mouse button (click, drag, and release).

Notice that the point on segment *BF* moves when the value of # is changed.

Another way is to click the arrows at either end of the bar. Each click moves the scroller exactly 1/100 of the way from one end of the bar to the other. The end positions initially represent # = 0.0 and # = 1.0, but this range can be reset. Another approach is to type the desired value for # into the edit box and press *Enter*. Finally, the scroll bar can be put into autopilot mode, by clicking either **autoreverse** or **autocycle**. If you do, the dialog temporarily disappears while the scrolling goes on automatically. The drawing window's title bar now tells you that you should press *Q* to halt the animation and restore the scrolling dialog.

Because many parts of a diagram can all be made to depend on the variable #, this method *synchronizes* their movement, all controlled by a single scroll bar. This can be very effective.

The program has actually been taught to recognize *three* symbols as numerical variables. In addition to #, there are also $ and @, each with its own **Animate/slide** control.

Click **Edit/Delete all**, and respond **No** when asked if the file is to be saved. The drawing window should be cleared. To start a new figure, click **Shape/Random/Triangle**. Triangle *ABC* should appear on the screen. Click **Point/Segment division**, type # into the *coordinate* box, type the whole list *AB*,*BC*,*CA* into the *segment* box (lower-case letters are fine, by the way), and click **Mark at**. Points *D*, *E*, and *F* should appear on segments *AB*, *BC*, and *CA*, respectively. (By the way, observe that there is a difference between typing *CA* and typing *AC* in this example.) Click **Line/Segments**, type the list *DE*,*EF*,*FD* into the box, and press *Enter* (or click **OK**). Triangle *DEF* appears. Click **View/Highlights/Polygon fill**, type *DEF* into the polygon box, and click **Fill** (or press *Enter*). The interior of triangle *DEF* should be filled with solid red. Click **Empty** to make the filling disappear. Click twice on the **Color** button, then click **Fill** again. Triangle *DEF* is now blue. Press *Escape* to close this dialog box. Open the **Measurement** dialog box, type *DEF/ABC*, and press *Enter*. Press *Escape* to close the dialog box. The ratio of the area of triangle *DEF* to the area of triangle *ABC* is now displayed in the drawing window. Slide the **Animate/# slide** scroll bar, and watch what happens.

To start a new diagram, you can click **Edit/Delete all** again, or you can click **2-Dim** to open a new drawing window. Do one or the other now to get a blank window. Click **Shape/Segment/OK**, which puts a simple segment *AB* on the screen. Click **Line/Angles/New**, type *AB* into the *initial ray* box (unless it is already there), type 360# into the *angle size* box (instead of using the mouse, you can use the *Tab* key to move the cursor from one box to the next), and press *Enter*. Press *Escape* to close this dialog box. Press *Ctrl+W* to show more of the figure, which consists of segment *AB* and ray *AC*. Click **Point/Segment division**, type *AC* into the *segment* box, # into the *coord* box, and click **Mark at**. Point *D* should appear on ray *AC*. Press *Ctrl+E* seven or eight times to enlarge the viewport. Now slide the **Animate/# slide** scroll bar and watch what happens. To see the *path* followed by point *D*, click **Animate/Tracing/new**, check that *D* shows in the *pen on* box, and press *Enter* (or click **OK**). The spiral path should appear, with your construction superimposed on it. Slide the scroll bar some more.

Click **Other/Lists/History**. This will open a text window that displays all the steps that have been used to create the current figure. When you are done reading the text, close the window by double-clicking the upper left corner.

Click **View/Grid** to turn on the graph paper background. Click **Btns/Coords** to change the mouse mode. Point at one of the four labeled vertices and hold down the left button. The coordinates of that point should appear, staying visible until you release the button.

Time for a fresh example. Get a new drawing window by clicking **2-Dim**. Click **Shape/Random/Convex polygon**, type 4 into the edit box, and press *Enter*. Quadrilateral *ABCD* should appear. Fill the interior with red by clicking **View/Highlights/Fill polygon**, typing *ABCD* into the *polygon* box, and clicking **Fill**. This dialog box must be closed, so press *Escape*. Click **Transf/Translate**, type *AC* into the *vector* box, and press *Enter*. A second red quadrilateral appears, the result of sliding *ABCD* along one of its diagonals. Click **Transf/Translate** again, type *BD'* into the *vector* box, and press *Enter*. Now you see four red quadrilaterals. Press the *F7* key (which is a shortcut for clicking **Transf/Again**) to get two more red quadrilaterals, then F7 again to run the total to eight. Now change the slide direction: Click **Transf/Translate**, type *DB'* into the vector box, and press *Enter*. There are now sixteen red quadrilaterals, and pressing *F7* twice runs the total to thirty-two, which is *enough*. Stop. There are far too many labels on the screen, so click **View/Labels/Hide**, or use the keyboard shortcut *Ctrl+L*, to turn the labels off. Click **Btns/Drag point** to change the mouse mode, then use the left button to drag points *A*, *B*, *C*, and *D*. Try to understand why the rest of the diagram reacts the way it does. Notice also what happens when you try to drag any of the other vertices.

This is an interesting and slightly complicated construction, so it is worth saving your work. This allows you to retrieve it later whenever it is wanted. Click **File/Save**. You will be asked to supply a file name. One name that comes to mind is *checkers*, but choose anything you want (up to eight characters), and press *Enter*. Notice that your filename now appears on the window’s title bar.

As a reminder that the whole diagram was generated by the four vertices of the quadrilateral in the upper left corner, mark them with dots for future reference. Right-click each of the four vertices twice each, which puts open circles where the labels used to be. These are the vertices that can be dragged to change the diagram. Now click **File/Save** again, to save the changes just made. Because you have already chosen a filename, the new save is immediate.

To see whether the file has truly been saved, click **2-Dim** to open another drawing window, then click **File/Old**, find the filename you chose in the list, select it, and press *Enter*. The diagram should appear just as it does in the other window. Close this drawing window, because it is not needed. You might as well close the other windows, too, because this tour is over.