

# Computer Programming with Logo



[www.mcweeney.co.u](http://www.mcweeney.co.u)

**Steve McWeeney**

## Licence Terms

You may print and photocopy unlimited copies of this booklet for use within a single educational institution.

You may, if you wish, charge students within the educational establishment for printed copies or CDs of the booklet. Neither the printed copies nor copies of the original pdf may be sold or otherwise traded outside the licensed institution.

You may not use the materials for personal profit.

The pdf file may be made publically available, in its original format, on a public network. It can be downloaded from the author's website.

Copyright is retained by the author.

The author can be contacted at [www.mcweeney.co.uk](http://www.mcweeney.co.uk)

Steve McWeeney

# Unit 1: Logo Commands

## Computer Languages

A computer language is made up of words or commands that give the computer instructions. There are many different computer languages. One of them is called LOGO. This language has been designed to draw graphics on the computer screen.

The graphics are drawn by instructing the computer to move an arrow on the screen. The arrow is called a turtle because sometimes LOGO is used to move a real device, which is called a turtle. As the turtle moves it draws lines.

## Some LOGO commands

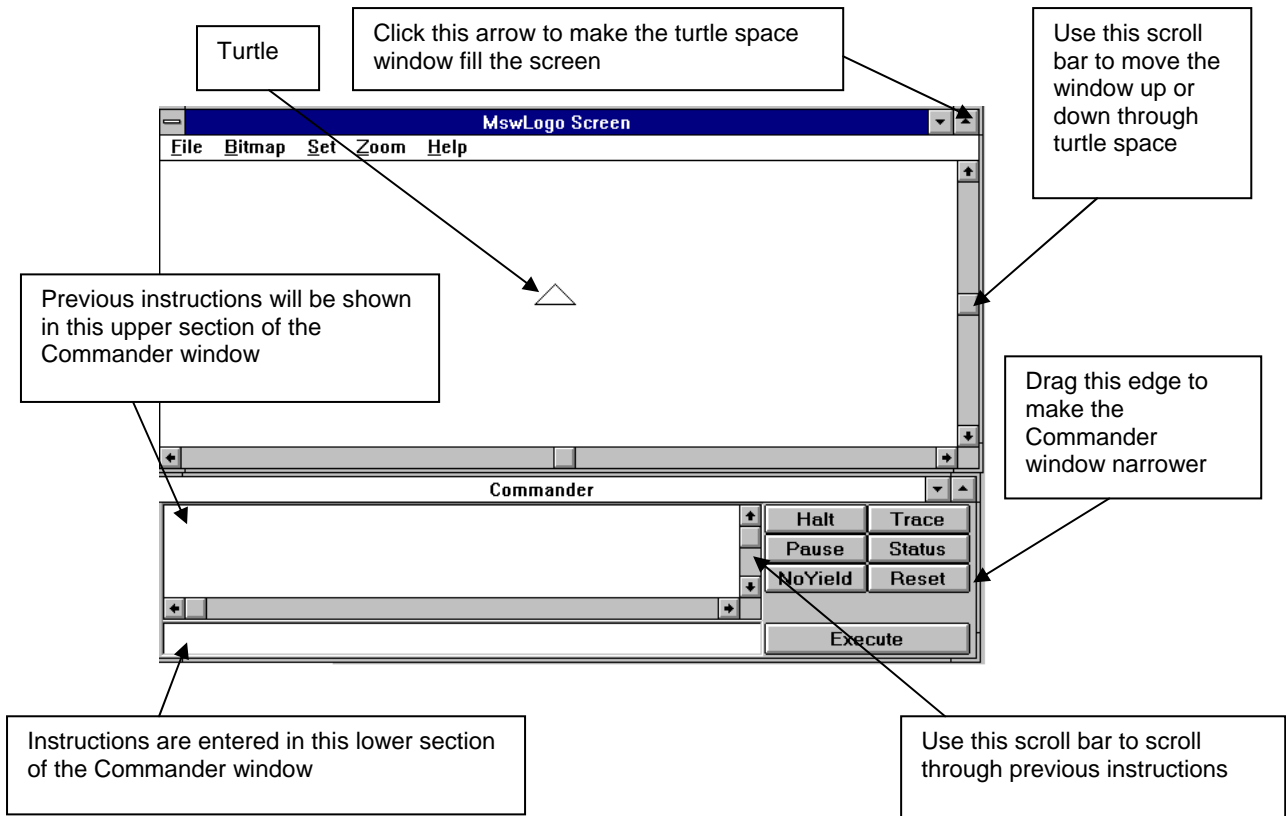
The first four LOGO commands that you will learn allow you to move the turtle backwards and forwards, to clear the screen and to return the turtle to its home position. They are given in the table below.

Command	What it does	Example
Forward	Moves the turtle forward a set distance . The example gives the command to make the turtle move forward 100 units.	Forward 100
Back	Moves the turtle back a set distance. The example shows the command to make the turtle move back 20 units	Back 20
Home	Moves the turtle back to its home position which is pointing upwards at the centre of the screen	
Clearscreen	Clears any lines that have been drawn and takes the turtle back to the home position	
Left	Turns the turtle left through a set angle. The example gives the command to make the turtle turn left through 90°.	Left 90
Right	Turns the turtle right through a set angle. The example gives the command to make the turtle turn right though 30°.	Right 30
Penup	This instruction stops the turtle drawing. It can still move about on the screen but it does not draw a line as it moves	
Pendown	This starts the turtle drawing again after it has been stopped by a Penup instruction.	

## Using LOGO

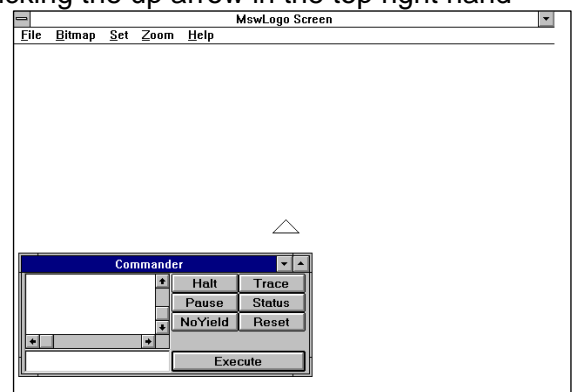
You are going to use a version of LOGO called MSWLogo. When you start up this program you will see two windows. The top window shows the turtle space – the area that the turtle moves around in. The turtle should be at the centre of this space, pointing upwards.

The second window is called the commander window. The commander window has two sections. The bottom section is where you enter commands. When the commands are entered and carried out, they are transferred to the other window. This allows you to scroll back through the commands that you have given.



## Practical 1

- (1) Go to a computer and log on. Start up the logo program by double clicking the MswLogo icon. Check that you recognise the different part of the screen as described above.
- (2) Make the turtle space window fill the screen by clicking the up arrow in the top right hand corner of the window. Your computer screen should now look like the diagram.
- (3) Enter the following four instructions  
Forward 100  
Right 90  
Forward 100  
Home
- (4) Log off





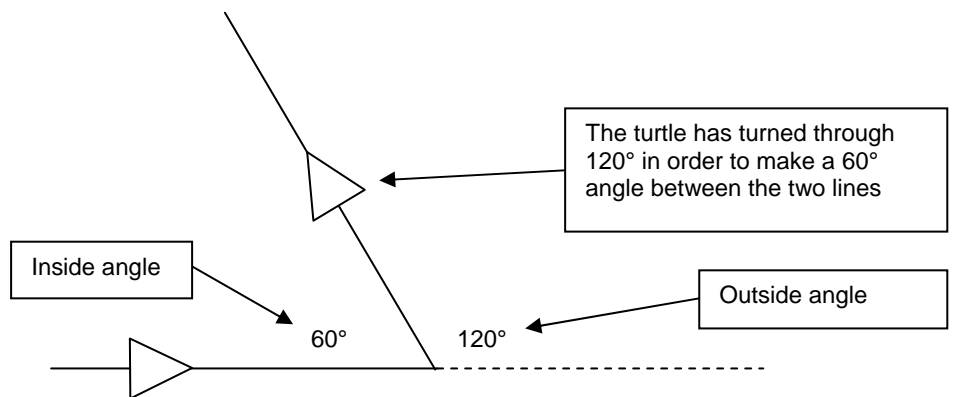


## Unit 2: Polygons and Loops

### The Turning Angle

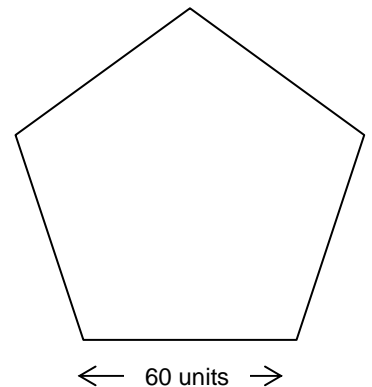
If you want to make the turtle draw two lines that have a particular angle between them, it is easy to mistake which angle you need to turn the turtle through. If you draw a diagram then it is often easier to work out the correct angle.

The diagram shows that, to draw two lines at  $60^\circ$  to each other, you must make the turtle turn through  $120^\circ$  because the angles on a straight line add up to  $180^\circ$ . It is important to realise that the turtle must turn through the outside angle, not the inside angle.



### Practical 3

- (1) Put the heading Pentagon in your exercise book.
- (2) A pentagon is a five sided figure with all the sides the same length and an angle of  $108^\circ$  between the lines. Write out the instructions needed to make the turtle draw a pentagon with each side 60 units long.
- (3) Have your work checked and then go to a computer and make the turtle draw a pentagon.
- (4) Print out your pentagon, cut it out and stick it underneath your set of instructions.



## Regular Polygons

A regular polygon is a closed figure where each side is same length and all the angles are equal. The diagram shows eight regular polygons.

The outside angle of a regular polygon can be found by dividing  $360^\circ$  by the number of sides.

The outside angle is the one that the turtle has to turn through.

$$\text{outside angle} = \frac{360^\circ}{\text{number of sides}}$$

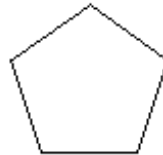
Name	Sides	Turtle turning angle
Triangle	3	$120^\circ$
Quadrilateral	4	$90^\circ$
Pentagon	5	$72^\circ$



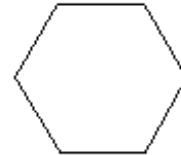
REGULAR TRIANGLE



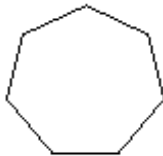
REGULAR QUADRILATERAL



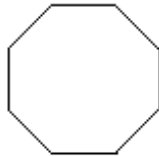
REGULAR PENTAGON



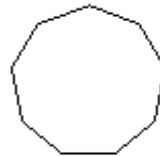
REGULAR HEXAGON



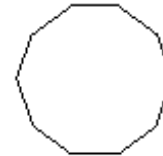
REGULAR SEPTAGON



REGULAR OCTAGON



REGULAR NONAGON



REGULAR DECAGON

## Logo Loops

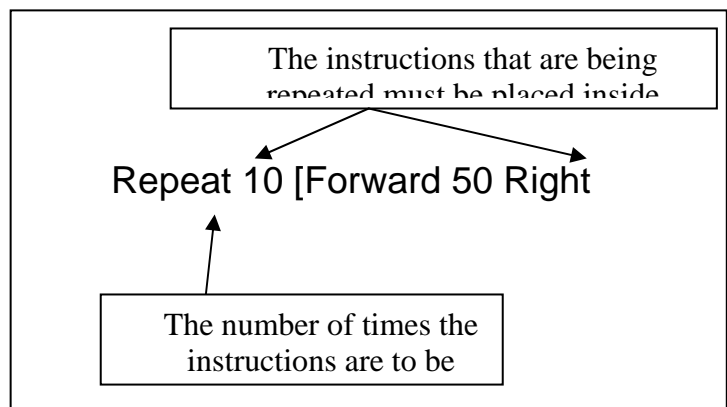
If we wanted to draw a regular decagon (ten sides) with a side of 50 units, we would need twenty turtle instructions. They would start off like this:

```
Forward 50
Right 36
Forward 50
Right 36
.....
```

the same movement and turn repeated ten times. Logo has a way of letting us tell the turtle to repeat something without having to type it out over and over again. This is the repeat loop.

The diagram shows the loop that would draw a regular decagon.

Notice how the loop instruction is built up. It starts with the word Repeat. Next comes the number of times that the instructions must be repeated. Finally, in square bracket, we have the instructions that will be repeated. You can put as many instructions as you like inside the square brackets.



## Practical 4

- (1) Go to a computer, log on, start up logo.
- (2) Type in the loop instruction for the decagon with a 50 unit side. This was given on the previous page.
- (3) When you have checked that you loop works, clear the screen.
- (4) Now work out the loop instruction to produce a regular octagon (eight sides) with each side 60 units long. Type this in to make the turtle draw your octagon.
- (5) Print off your octagon and write the loop on the printout. Take your printout to your teacher to have it marked

## Writing 2

Copy the writing from this box into your exercise book. Don't forget to underline the heading and put the date in. Fill in the missing words

### **Loops**

We can tell the turtle to repeat a set of instructions any number of times using a Loop. We start a loop with the word                               . After this we put a number. This is the number of times that the loop is to be done. Next we put the instructions that must be repeated inside                          brackets. The following loop would draw a regular octagon with each side 60 units long:

Repeat      [                                         60 Right           ]



# Procedures

## Procedures

Up to now you have entered instructions for the turtle to follow and it has moved straight away. A procedure allows you to enter a whole sequence of instructions ready for the turtle to follow when you want.

An advantage of this is that you only have to write out the instructions once. Then, every time you want the turtle to follow these instructions, you tell it to do the procedure.

Another advantage is that, if there is a mistake in the instructions, you can go back and change the procedure. You don't have to write out all the instructions again.

Logo procedures start with the word To. The name of the procedure comes after the To. You have to make up a name for a procedure and it is best to choose a name that will help you remember what the procedure is for. The last line of the procedure must be the word End.

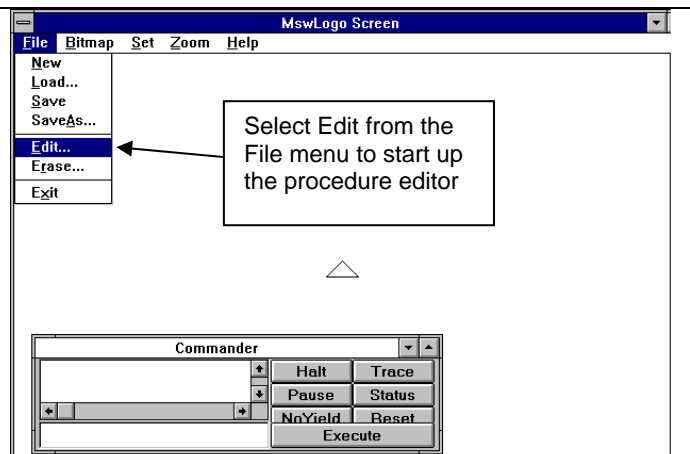
When you want the turtle to follow the instructions you just type in the name of the procedure.

## Using the Editor

Logo uses a special editor to let you type in procedures and, if necessary, change them later.

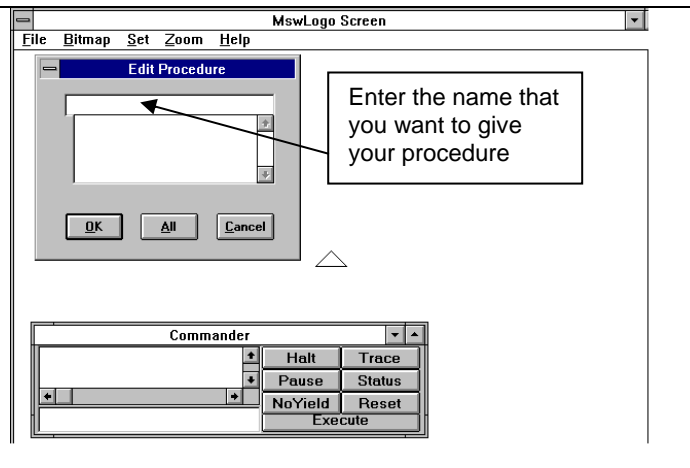
### Step 1

Select Edit from the File menu on the logo screen



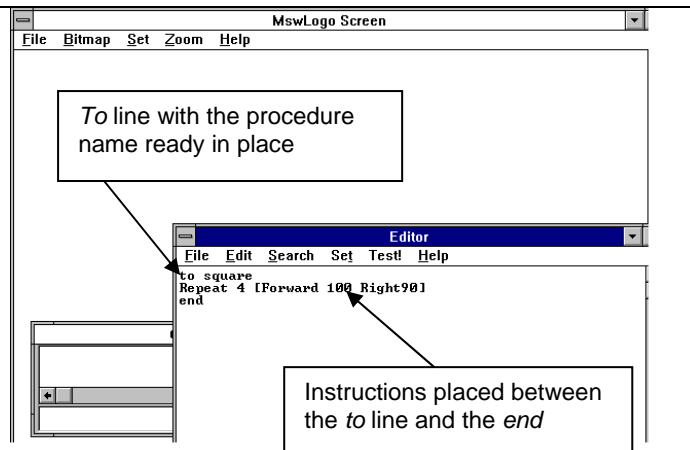
### Step 2

Enter the name that you want to give your procedure. You cannot use spaces in a procedure name. When you choose a name for a procedure you should try to make the name match what the procedure does. For example, if you write a procedure to draw a square, call the procedure square.



### Step 3

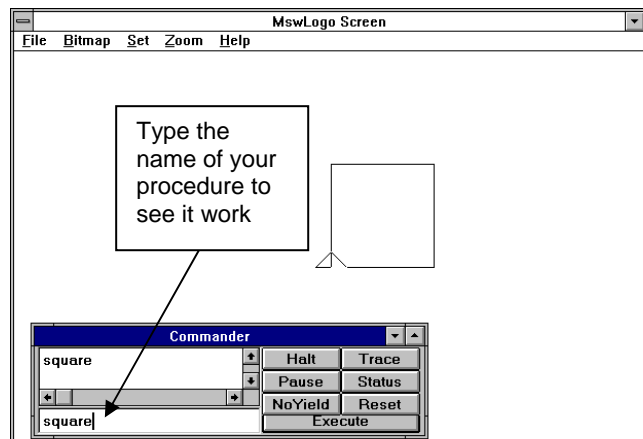
The Editor window will open. The first line and last line of your procedure will be ready done for you. Type in the instructions for your procedure between these two lines. Do not change the to or the end line. You can have as many lines of instructions as you want.



### Step 4

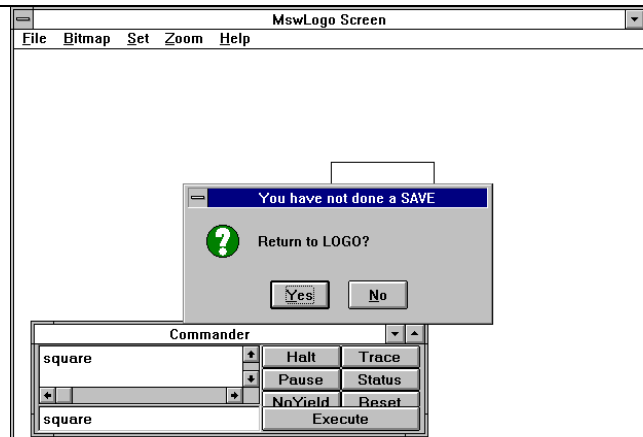
Use File and Exit from the Editor window to leave the editor. You will be asked if you want to save the changed version. Say yes.

You can now make your procedure work by typing in the name of your procedure. You can think of writing a procedure as teaching the turtle a new word.



### Step 5

You must save your work before leaving logo. If you don't then you will get this error message. Click Yes to return to logo and then use Save from the File menu to save your work. Next time you use Logo you will be able to load your procedures back in.



## Practical 5

- (1) Go to a computer, log on and start up Logo.
- (2) Follow the steps given above to enter a procedure that draws a square of side 100 units. Call your procedure square.
- (3) Test your procedure. Once you have it working call your teacher to see it on the screen so that your practical log can be signed.
- (4) Save your work. Call it patterns.lgo (Note that this program is the only one that you use where you have to put the full stop and last three letters in for yourself). Your file name must end with the full stop and the letters lgo
- (5) Log off.

## Writing 3

Copy the writing from this box into your exercise book. Don't forget to underline the heading and put the date in. Fill in the missing words

### **Procedures**

A sequence of instructions written for the turtle to follow at a later time is called a \_\_\_\_\_. The first line of a procedure starts with the word \_\_\_\_\_ followed by the name we want to give to the procedure. The last line of a procedure must be the word \_\_\_\_\_.

We use a special \_\_\_\_\_ program to write a procedure and to change any mistakes in it. We make the turtle follow the instructions in the procedure by typing in the \_\_\_\_\_ of the procedure.

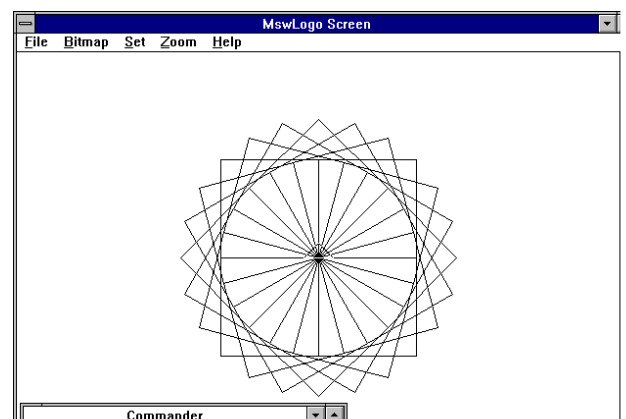
## Practical 6

- (1) Go to a computer, log on and start up Logo.
- (2) Load in the work you saved last time. It should be called patterns.
- (3) You are going to write a procedure called pattern. This procedure will draw a square and then turn the turtle through  $15^\circ$  and draw another square. It will do this 24 times ( $24 \times 15^\circ = 360^\circ$ ). You already have a procedure called square that you can call from the new procedure. The new procedure will look like this

```
To pattern
Repeat 24 [ square Right 15]
End
```

Start up the editor and enter the new procedure. Save it as you close the editor down.

- (4) Run the new procedure to make sure that it works. You should end up with the pattern shown.
- (5) Print off your work and take it to your teacher to have your practical log signed.





# Unit 4: Variables and Numbers

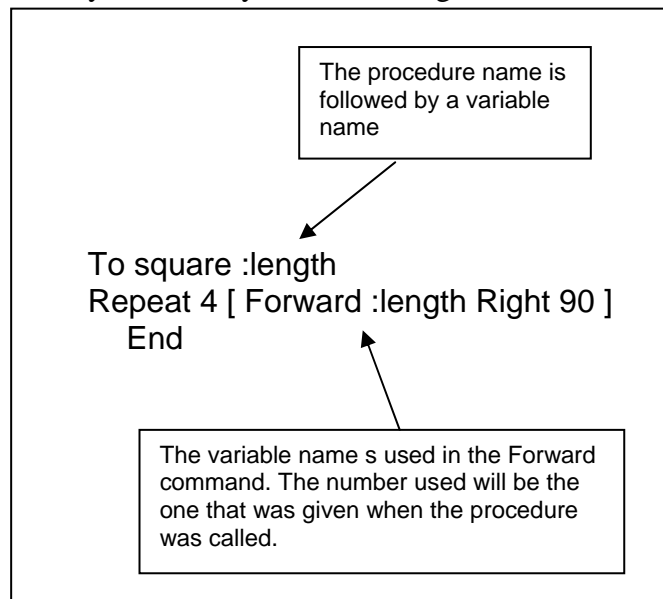
## Passing Numbers to Procedures

All the procedures that we have written so far always do exactly the same thing. Sometimes it is better to write a procedure that is more general and pass the details to it when we want to use it.

As an example of this, the procedure on the right draws a square. We tell the procedure how big the square will be when we call it. For example square 150 would draw a square of side 150 units.

We can now use this procedure to draw any size square.

Notice that the variable name starts with a colon (:). All variable names must start with a colon.



## More than one Variable

You can pass more than one variable in a procedure. Suppose we want to write a procedure that draws a any regular polygon. We want to tell the procedure how many sides the polygon should have and how long the sides should be. Here is a procedure that would draw the polygon.

```
To polygon :sides :length
If :sides <=3 then stop
Repeat :sides [Forward :length Right 360/:sides]
End
```

In this procedure, the variable `:sides` stores how many sides we want while the variable `:length` stores how long each side is to be. The two variables follow the procedure name.

The first line of the procedure stops the procedure if the number of sides is less than or equal to 2. This stops the procedure trying to draw a non-existent polygon.

The repeat loop needs to be done as many times as there are sides so the repeat loop starts with:

Repeat sides:

Inside the loop the turtle is moved forward whatever distance is stored in the `:length` variable. Then the turtle is turned right. Since the angle of turn depends on the number of sides, the turning angle is calculated by dividing 360 by the value stored in the variable `:sides`.

If we wanted to draw an octagon (eight sides) with each side 30 units long, we would enter:

Polygon 8 30

## Practical 7

- (1) Put a heading in your exercise book Using Variables.
- (2) Design a procedure that will draw a square of any size, starting anywhere on the screen. For example square 100 50 -150 should take the turtle 50 units to the right and 150 units down (from home) and then draw a square of side 100 units (Negative numbers make the turtle move back)
- (3) Go to a computer and test your procedure
- (4) Now use your procedure to design another procedure, called foursquare that draws four squares, one in each quarter of the screen.

# Unit 5 Assessment

There are a series of tasks given below. Each task depends on the one before it being completed. The more you complete the better your assessment level is likely to be. You will need to present all your design work, including written-out versions of your procedures, together with printout showing your procedures working.

You will gain credit for designing procedures that are efficient.

- (1) Turtle would like a capital T drawing. Design a procedure that will draw the letter with the dimensions shown.
- (2) Turtle would like to see what pattern you get when the T is drawn 20 times, rotating  $18^\circ$  each time. Design another procedure that will do this.
- (3) Turtle would like to be able to draw a T any size. Design a procedure that will draw a T shape given the length of the main rectangles. The width of the rectangles is always  $1/5$  of the length.

