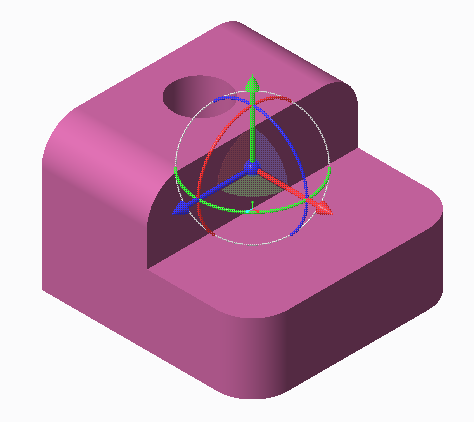
**Tutorial:** Basic G-Code Programming  
**Reference:** Coordinate Axis Direction



Y

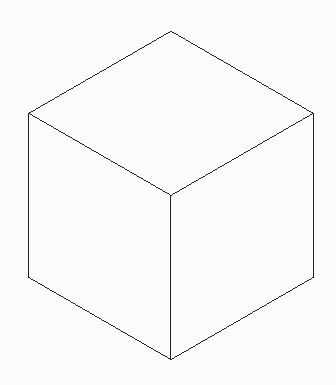
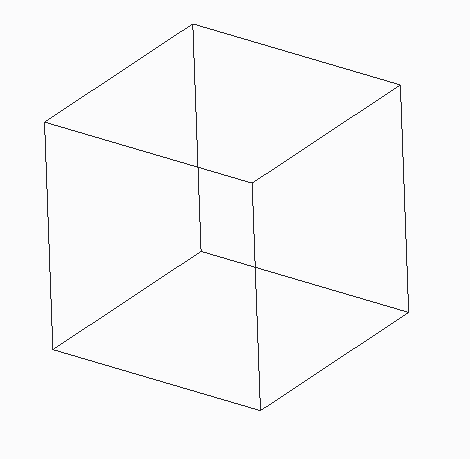
X

Z

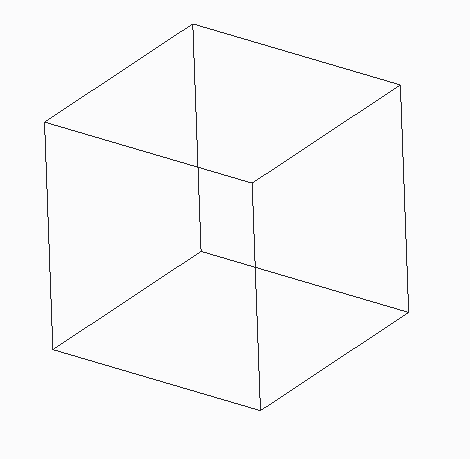
This tutorial is designed to create a wireframe model of a part. This tutorial will not simulate a full G-code program for a 3D print because the program for a whole object is, at minimum, 15,000+ lines long. This is because of the complexity of the layering and infill pattern.

Cube

10 mm x 10 mm x 10 mm

CAD Solid Model

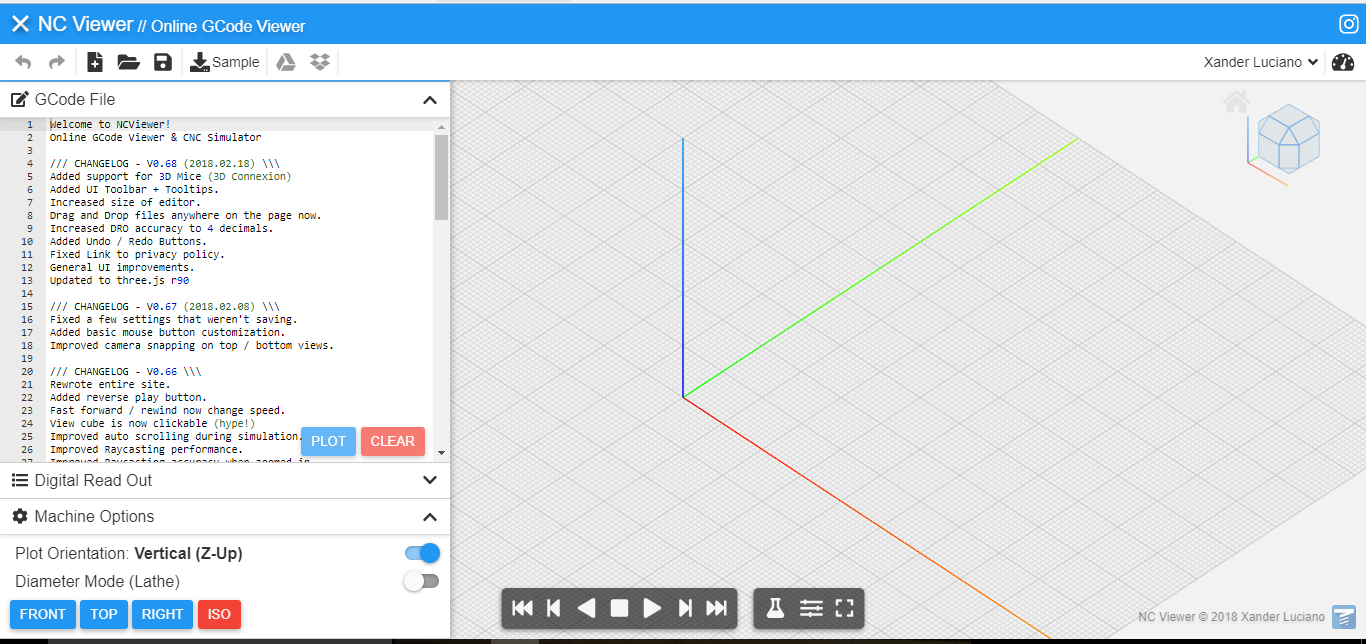
Wireframe Model 

**NC Viewer Program**

1. Open Browser > go to **ncviewer.com**   
   This site is a simulator for compiling machining code, whether it is machining, turning, or 3D printing.

Standard menu bar

Click surfaces to look at 2D views



Origin

Program area

Simulation work space

Runs code tracking to plot program

Quick access views

2D: front, top, and right

3D: Iso (isometric) default view

Downloads a .txt file

**Menu Bar**



Opens a sample program

Open program: file types: .gcode   
 .txt

New program

Undo and redo

**Work screen movement**

**Mouse**

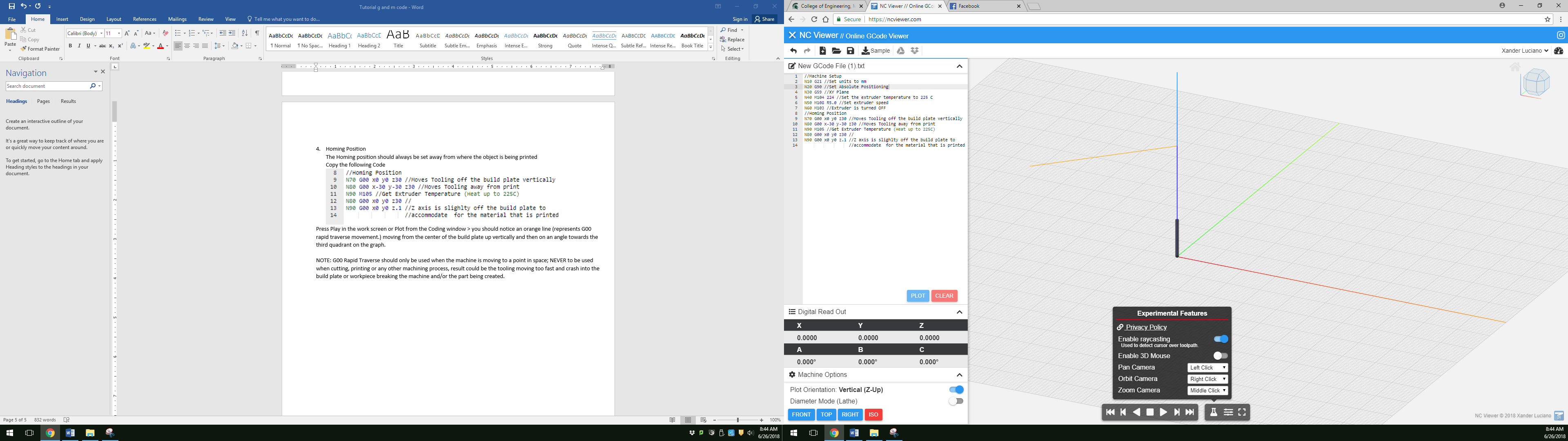
Left Click - Select

Left Hold - Pan

Right Hold - rotate screen

Center Roll Bar - Pull = Zoom out

Push = Zoom in



**Machine Setup Code**

**Setting Units**

G20 = inches

G21 = mm

It is important to set the units so the machine knows how far to incrementally change when programming in the distance from one point to the next

**Coordinate Position**

G90 = absolute position

G91 = incremental position

G92 = offset coordinate system

User needs to make allotment for the size of the tool they are using. Use absolute position make the change in the *x*, *y*, *z* coordinates when laying out the coordinate points. Use incremental or offset positioning within the program setup phase.

Note: Because we do not know the size of the tool we are using, absolute position will work best.

**Homing**

A series of *x*, *y*, *z* coordinate movements that moves the tool away from the build plate allows for easy setup and removal of the object. The home position is typically a location up and away from the build plate or off the stock material that is to be machined so nothing gets damaged when the extruder head heats up or the cutting tool begins its motion. For this tutorial we will set a home position that is to the upper-right of the wireframe model. The user will not set this position as the origin of the virtual machine, but keep the origin point as one of the corners of the object.

**Code: Cube 10 mm x 10 mm x 10 mm**

1. Select New Program Icon



1. Type in the following Code   
   (NOTE: Two forward slashes // allow the user to place a comment. The code after the “//” will not be recognized by the machine and therefore will not interfere with the code.



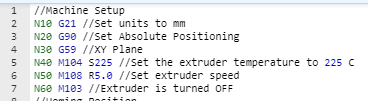
Coded line ID generated by user. This is designed to help identify a line of code when each line has a unique marker. Incremental changes (for example: N10, N20) does not have to be by tens, but can be setup as any incremental change. Creating a larger space between ID changes allows the user to add a new line between two existing lines; this way the rest of the program does not need to be renumbered.

Coded line ID generated by software. Note that not all compilers contain line code ID .markers

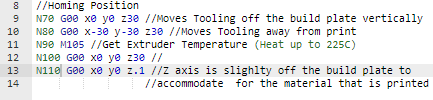
1. The next line of codes are used to set up the machine.
   1. 3D Printer: setting up extruder head temperature, build plate temperature, etc.
   2. CNC machine: tool number from the carousel, tool speed, etc.

We will setup this program as if we were using a 3D printer to create the object.

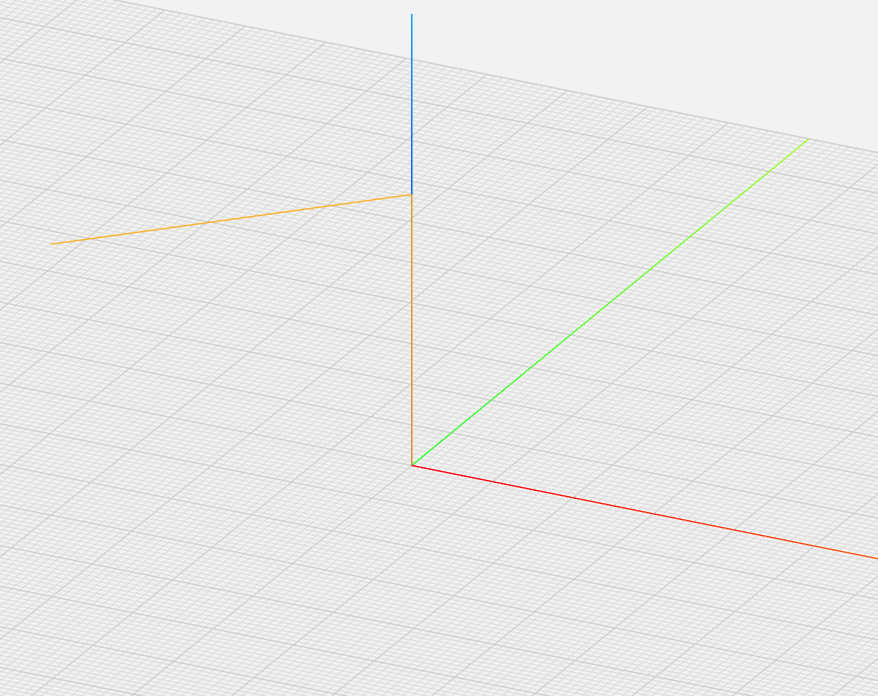
Practice writing the following code below:



1. The homing position should always be set away from where the object is being printed; continue writing the following code.



Press play in the work screen or plot from the coding window and you should notice an orange line moving from the center of the build plate up vertically and then on an angle towards the third quadrant on the graph.



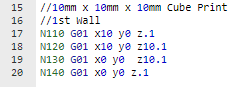
Home position

Once this position is reached the extruder will heat up then traverse back over the coded points to begin printing.

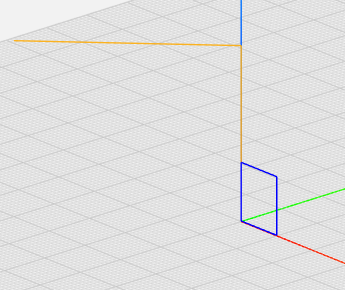
Movement 2 (Line: N80) and Movement 3 (Line N100)

Movement 1 (N70) and Movement 4 (110)

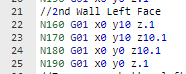
1. Code the cube 10 mm x 10 mm x 10 mm
   1. 1st wall (front surface): write down the following lines of code.

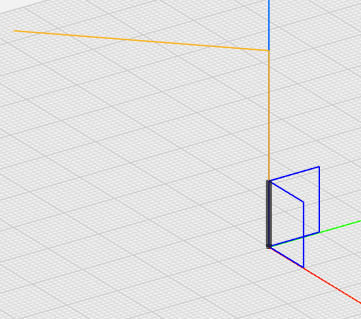


* 1. Press play to plot the code.



* 1. 2nd wall (left surface): write down the following lines of code. Press play to plot the code. Current tool location = (x0 y0 z.1)



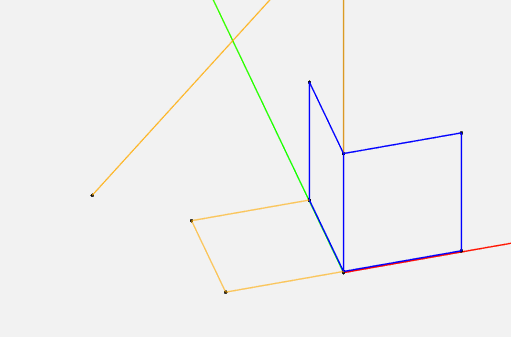


* 1. Reposition the tool:  
     Option 1: trace a line to move the tool back to starting location.



Option 2: Move the tool away from the part and reposition it using a combination of G00 and G01. For Example:





N200-N220 Movements to move the tool off the part and reposition it at desired location (Back Wall)

Front View (1st Wall)

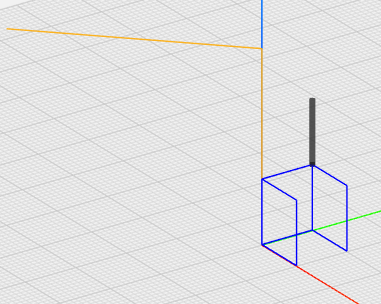
Left Side (2nd Wall)

Home Position

Note: For this tutorial user will follow Option 1

* 1. 3rd wall (back surface): write down the following lines of code. Press play to plot the code. Current tool location = (x0 y10 z.1)



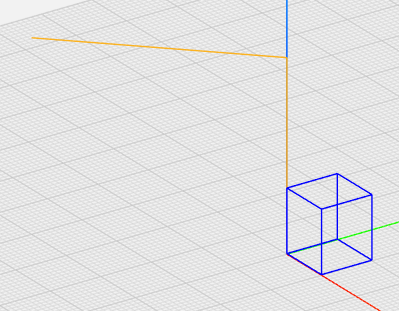


* 1. Trace a line to move tool back top right corner of back surface



* 1. 4th wall (right surface): write down the following lines of code. Press play to plot the code. Current tool location = (x10 y10 z10.1)





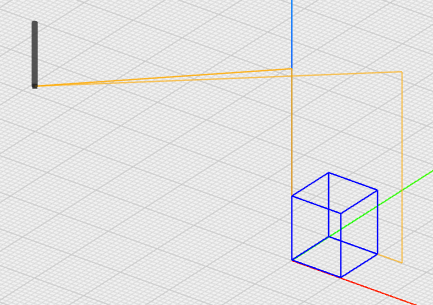
* 1. The top and bottom surfaces are complete based on the shared edges from the other four surfaces.

1. Return tool to the home position

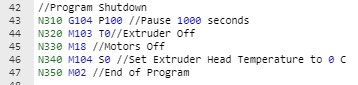
Once a part is complete it is important to return the tool back to its home position. This makes it easier to remove the part from the machine. The user should make sure not to code movements that run into the part, so we will move the tool away from the part with a series of moves.

Copy the following code > Plot Code





1. Machine shutdown

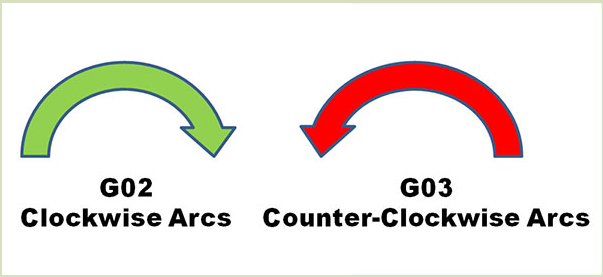


**Code: Curves, Arcs, and Circles**

To create curves, arcs, or circles, the user needs to choose one of two methods. Method 1: use the radius for its formation; Method 2: use offset values from the start point. For both methods the user needs to program the proper G-code axes (G17=XY, G18=ZX, G19=ZY) - this will place the tool at the correct orientation to create the curve, arc or circle based upon its view plane. Not every machine has the ability to rotate the tool into the different axis positions, so read the instruction manual of a particular machine before trying to run a series of code.

**Method 1 Using R (radius)**

Next we will modify the existing code by adding curves. There are two types of curved edges we can code:

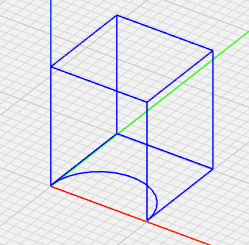


G02: Clockwise interpolation

G03: Counter-clockwise interpolation

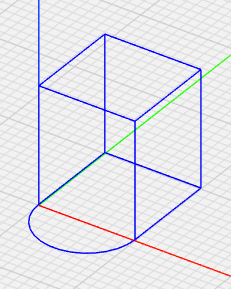
The direction of the curve (concave vs convex) is dependent upon the direction of the tool.

For example: tool moves in a counter clockwise direction (x0 y0 z.1 to x10 y0 z.1)



G02: Clockwise interpolation



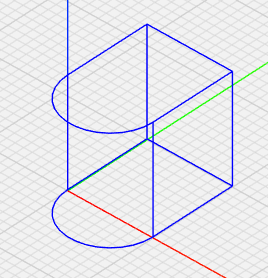


G03: Counter-clockwise interpolation



1. Change the cube code to the G03 counter-clockwise interpolation.
2. Change the following lines of code to create an arc on the top edge of the front surface; notice how this is a G02 (clockwise). This is because the tool is moving in a clockwise direction from line N130 to line N140.





**Method 2** Offset Values: I, J, K

Coding a curve, arc, or circle by radius does not always ensure the proper size in G-Code. Options using code I, K, J allow the user to check the location of curve, arc, or circle by setting values from the start point to locate the center of the object.

The following shows the relationship between axis (*x*, *y*, *z*) and offset value (I,J,K)

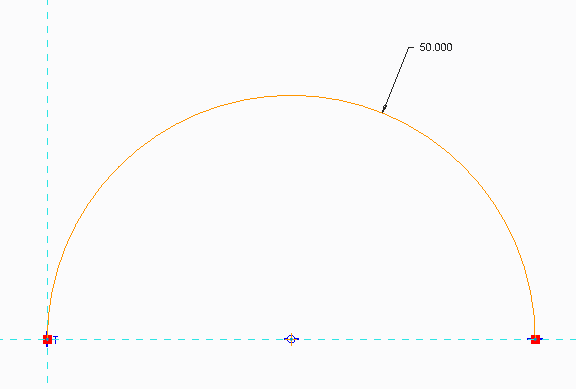
I = *x*-axis

J = *y*-axis

K = *z*-axis

Codes I, J, K represent the offset distance from the starting point for the curve, arc, or circle in the direction of motion. The I, J, K value can be either positive or negative based on the starting point and direction of the tool.

Y (J)



R50

Code  
G01 x0 y0 z0

G02 x50 y0 z0 I50

Start Point (0,0)

End Point (50,0)

X (I)

Y (J)

X (I)

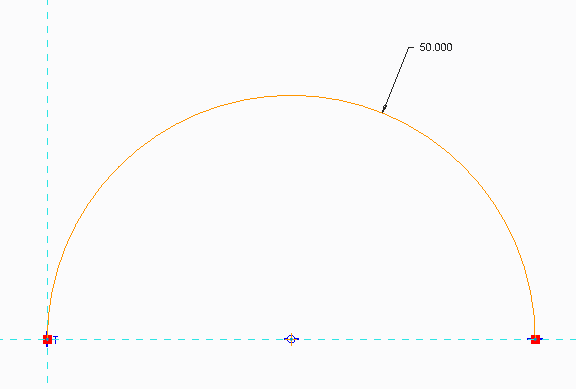
End Point (0,0)

Start Point (50,0)

Code  
G01 x50 y0 z0

G03 x0 y0 z0 I-50

R50



1. Change the Code to the following

//Cube Print 10mm x 10mm x 10mm

//1st Wall Front Face

N120 G03 x10 y0 z.1 I5 //I=X-Axis Radius 5 (-5 because of direction of tool; Tool is moving to the right from previous point making the directional motion negative

N130 G01 x10 y0 z10.1

N140 G02 x0 y0 z10.1 I-5 //I=X-Axis Radius 5 (-5 because of direction of tool; Tool is moving to the left from previous point making directional motion negative

N150 G01 x0 y0 z.1

**Extra assignment:** Add an arc of 5 mm that is 180 degrees on the right face of the cube. Decide the following: 1) choose *y* or *z* direction; 2) concave or convex arc.