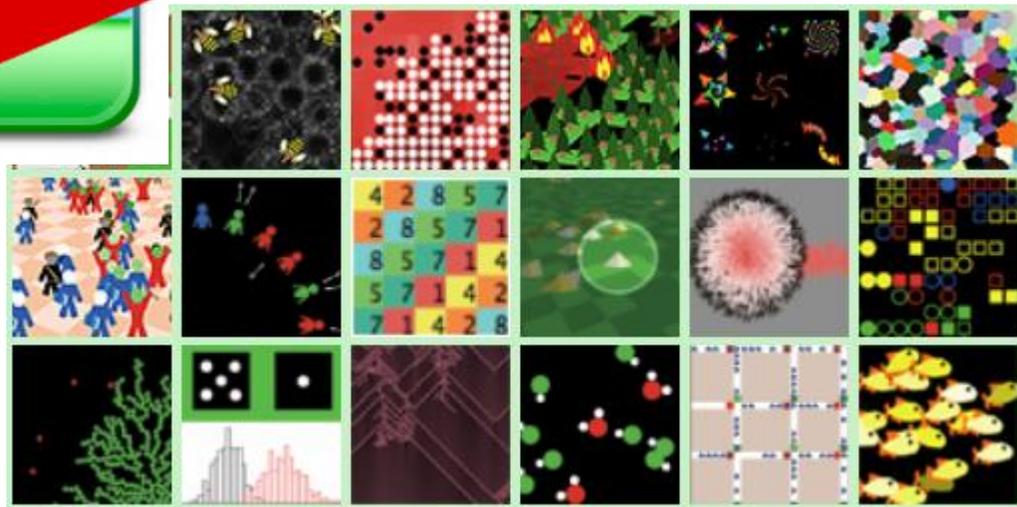




General Sir John Kotelawala Defense University
Faculty of Computing
Department of Computer Engineering

BSc (Hons) in Computer Engineering Practical Guide

NetLogo Simulation



1. INTRODUCTION: NetLogo

Nowadays, several different Agent-Based modeling languages are existing (Such as Any Logic, Ascape, etc.). Although, Netlogo remains the most widely used. NetLogo is a multi-agent programming language and modeling environment for simulating complex natural and social phenomena. The social system consists of individuals interacting with each other.

Uri Wilensky, the developer of NetLogo has conducted agent-based modeling research for over three decades. It is particularly well suited for modeling complex systems evolving. Modelers can give instructions to hundreds or thousands of independent “agents” all operating concurrently, to explore connections between micro-level behaviors of individuals and macro-level patterns that emerge from their interactions. NetLogo was designed for both education and research. It is freeware - anyone can download it for free and build models without restriction.

1.1. Menu Guide

We can find common menu items such as File, Edit, etc. the same as other software. The menu bar is found at the top of the NetLogo window.

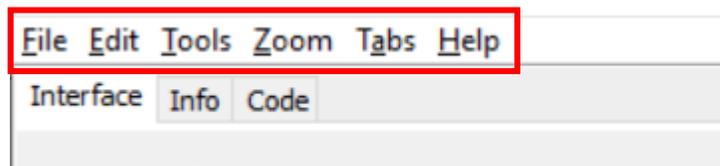


Figure 1.1, Menu Bar

File

From the file menu....

Table 1.1: Description of the components of the File menu

New	Starts a new model.
Open	Opens any NetLogo model on your computer.
Models Library	A collection of demonstration models.
Recent Files	Re-opens any previously model opened with "File -> Open"
Save	Save the current model, or the currently selected source file.
Save As NetLogo Web	Save the current model, or the currently selected source file, using a different name.
Save All	Save the current model and all open source files. This option is only available when one or more source files are open.
Upload to Modeling Commons	Uploads the model to Modeling Commons. See Modeling Commons
Save As NetLogo Web	Saves a web page, in HTML format, containing NetLogo web running your model.

Export World	Saves all variables, the current state of all turtles and patches, the drawing, the plots, the output area, and the random state information to a file.
Export Plot	Saves the data in a plot to a file.
Export All Plots	Saves the data in all the plots to a file.
Export View	Save a picture of the current view (2D or 3D) to a file (in PNG format).
Export Interface	Save a picture of the current Interface tab. (in PNG format)
Export Output	Save the contents of the output area or the output section of the command center to a file.
Export Code	Save the model's code to an HTML file, preserving colors.
Import World	Load a file that was saved by Export World.
Import Patch Colors	Load an image into the patches; see the import-pcolors command.
Import Patch Colors RGB	Load an image into the patches using RGB colors; see the import-pcolors-rgb command.
Import Drawing	Load an image into the drawing, see the import-drawing command.
Import Hub Net Client Interface	Load the interface from another model into the Hub Net Client Editor.
Print	Sends the contents of the currently showing tab to your printer.
Quit	Exits NetLogo. On a Mac, this item is on the NetLogo menu instead.

As mentioned in the above table we can retrieve any demonstration models from the master collection using the search bar.

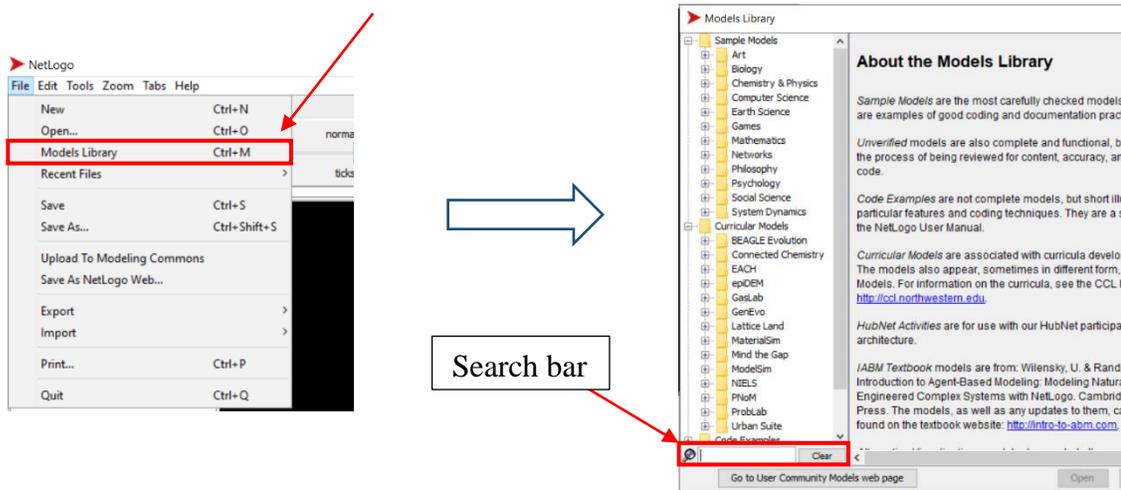


Figure 1.2, Model Library

Edit

From the edit menu...

Table 1.2, Description of the components of the Edit menu

Undo	Undo last text editing action you performed.
Redo	Redo last undo the action you performed.
Cut	Cuts out or removes the selected text and temporarily saves it to the clipboard.
Copy	Copies the selected text.
Paste	Places the clipboard text where the cursor is currently located.
Delete	Deletes selected text.
Select All	Select all the text in the active window.
Find...	Finds a word or sequence of characters within the Info or Code tabs.
Find Next	Find the next occurrence of the word or sequence you last used to Find with.
Comment /	Used in the Code tab to add or remove semicolons from code (semicolons are used in NetLogo code to indicate comments).
Uncomment	
Shift Left / Shift Right	Used in the Code tab to change the indentation level of code.
Format	Used in the Code tab to correct the indentation of the currently selected code.
Snap to Grid	Available only in the Interface Tab. When enabled new widgets stay on a 5-pixel grid so it is easier to line them up. (Note: this feature is disabled when zoomed in or out.)
Convert from 5.3.1 to 6.0	Available only in a ".nls" code tab. Treats the code in this tab as though it were written in 5.3.1 and converts it to run in NetLogo 6.

Tools

From the tools menu...

Table 1.3, Description of the components of Tools menu

Preferences	Opens the preferences dialog, where you can customize various NetLogo settings.
Extensions	Opens the Extension Manager, where you can discover and manage Extensions.
Halt	Stops all running code, including buttons and the command center.

Global Monitor	Displays the values of all global variables.
Turtle Monitor	Displays the values of all of the variables in a particular turtle.
Patch Monitor	Displays the values of all of the variables in a particular patch.
Link Monitor	Displays the values of all of the variables in a particular link.
Close All Agent Monitors	Closes all open agent monitor windows.
Close Monitors for Dead Agents	Closes all open agent monitor windows targeting dead agents.
Hide/Show Command Center	Makes the command center visible or invisible.
3D View	Opens the 3D view. See the Views section of the Interface Tab Guide for more information.
Color Swatches	Opens the Color Swatches. See the Color Section of the Programming Guide for details.
Turtle Shapes Editor	Draw turtle shapes. See the Shapes Editor Guide for more information.
Link Shapes Editor	Draw link shapes. See the Shapes Editor Guide for more information.
Behavior Space	Runs the model over and over with different settings.
System Dynamics Modeler	Opens the System Dynamics Modeler. See the System Dynamics Modeler Guide for more details.
Preview Commands Editor	Allows for easy editing of the commands sequence used to create preview images for models.
Hub Net Client Editor	Opens the Hub Net Client Editor. See the Hub Net Authoring Guide for more details.
Hub Net Control Center	Disabled if no Hub Net activity is open. See the Hub Net Guide for more information.

Under the tools menu, the color swatches provide colors in different ways. A color can be number in the range 0 to 140, except 140 itself. We can define the color of an agent or patch used by the word of the color or by the number shown in the chart below, the chart showing the range of such colors you can use in NetLogo. On the bottom right there are three increment options, 1, 0.5, and 0.1 as highlighted in the given figure 1.3. These numbers indicate the difference between two adjacent swatches. When the increment is 1 there are 10 different shades in each row; when the increment is 0.1 there are 100 different shades in each row. 0.5 is an intermediate setting.



Figure 1.3, Color Swatches

Turtle Shape Editor

To begin making shapes, choose Turtle Shapes Editor or Link Shapes Editor in the Tools menu. The Shapes Editor allows you to edit shapes, create new shapes, and borrow from another model. You can also import turtle shapes from a library of pre-existing shapes.

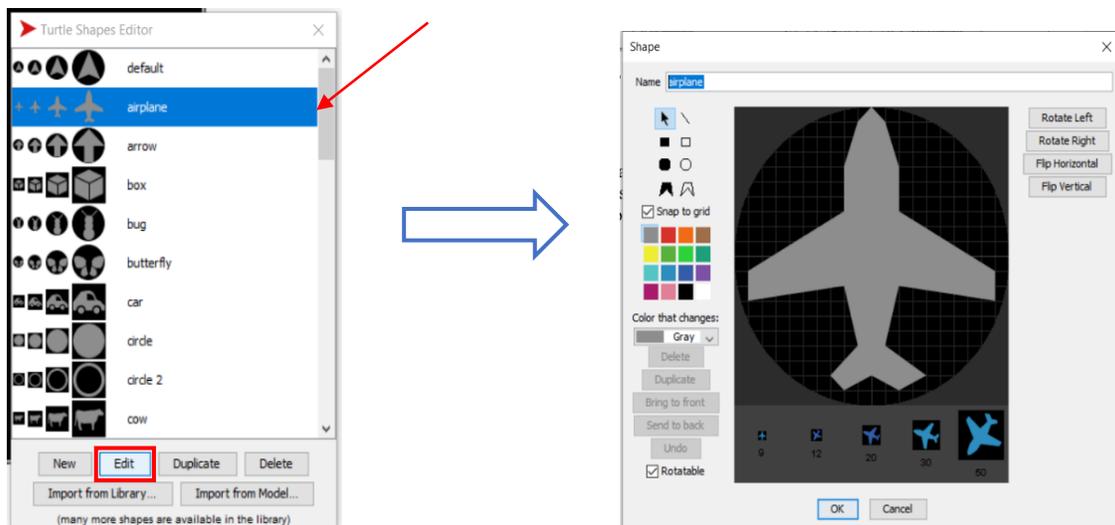


Figure 1.4, Turtle Shape Editor

Help

From the help menu...

Table 1.4, Description of the components of the Help menu

Look Up in Dictionary	Opens a browser with the dictionary entry for the selected command or reporter
NetLogo User Manual	Opens this manual in a web browser.
NetLogo Dictionary	Opens the NetLogo Dictionary in a web browser.
NetLogo Users Group	Opens the NetLogo Users Group site in a web browser.
Introduction to Agent-Based Modeling	Opens the MIT Press page for "Introduction to Agent-Based Modeling"
Donate	Opens the NetLogo donation page in a web browser.
About NetLogo 6.1.1	Information on the current NetLogo version you are running.

NetLogo Dictionary

This helps to retrieve the information and the code for any particular keywords with some examples.

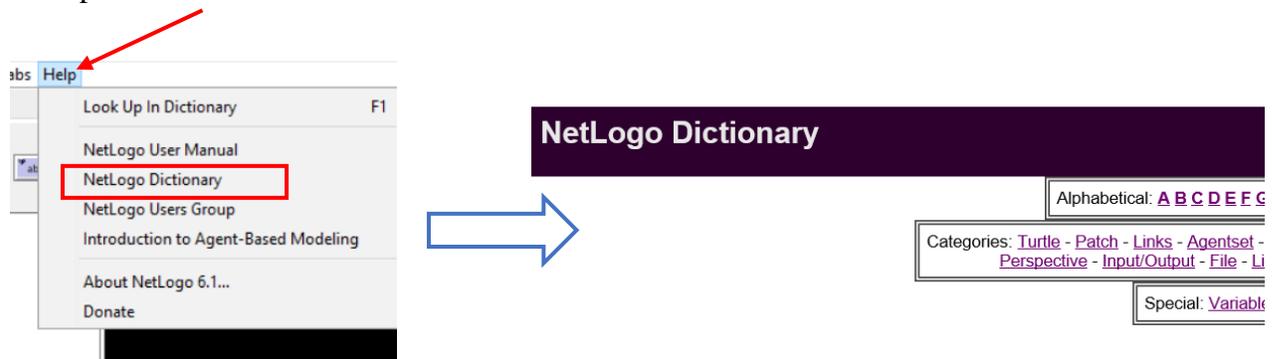


Figure 1.5, Dictionary

1.2. User Interface Guide

At the top of Net Logo's main window are three tabs labeled "Interface", "Info" and "Code". Only one tab at a time can be visible, but you can switch between them by clicking on the tabs at the top of the window.

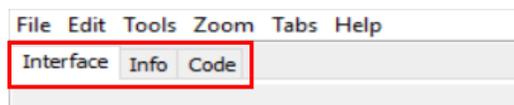


Figure 1.6, Tabs

Interface Tab

On the right is the graphics window, in which the “world” of the model is made visible after opening the “Interface”. This is the place to implement our environment.

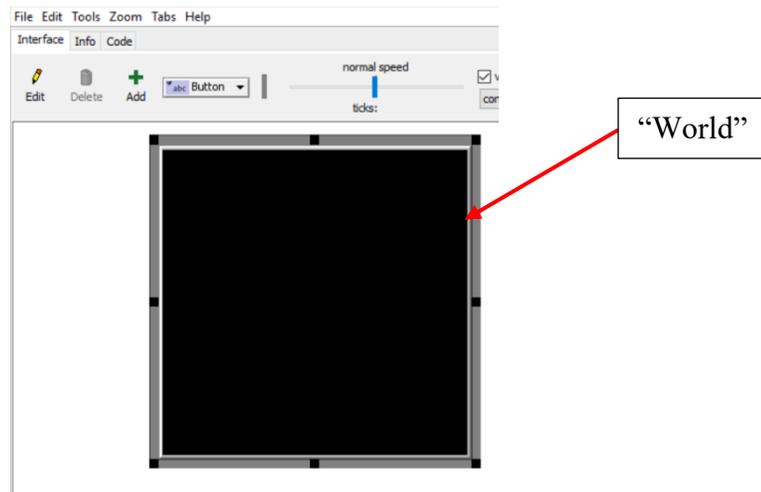


Figure 1.7, Interface

There is several editing to organize our environment (Right-click on the world → Edit)

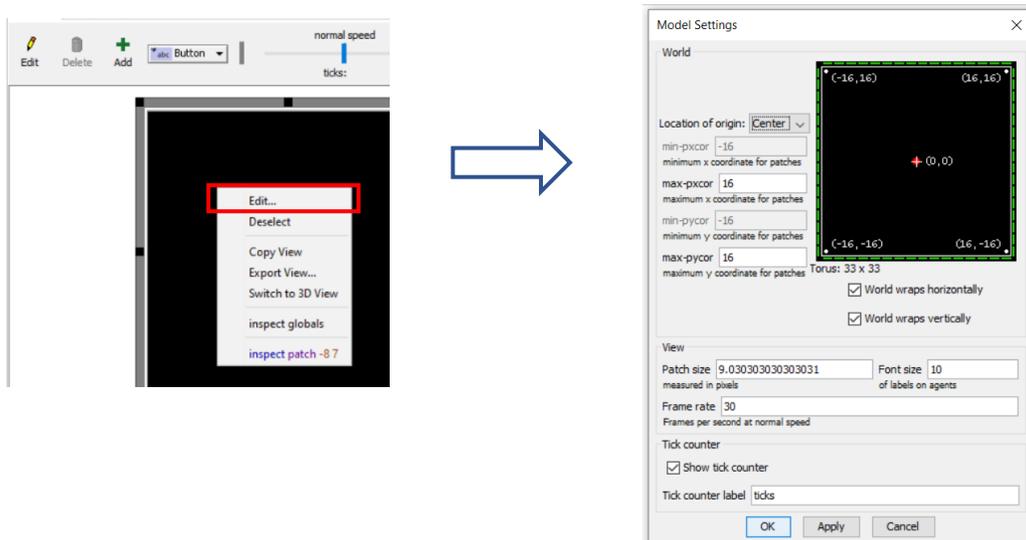


Figure 1.8, Edit the World

As shown in the above figure 1.8, we have to provide the value for pxcor, pycor, Patch Size, Font Size, and Frame rate to set up the world. Also, the world is called “Patch”.

pxcor: - Patch X-Coordinate (minimum & maximum)

pycor: - Patch Y-Coordinate (minimum & maximum)

Info Tab

The Info tab provides an introduction to a model. It explains what system is being modeled, how the model was created, and how to use it, and so on. We can edit the information with the edit option as shown below.

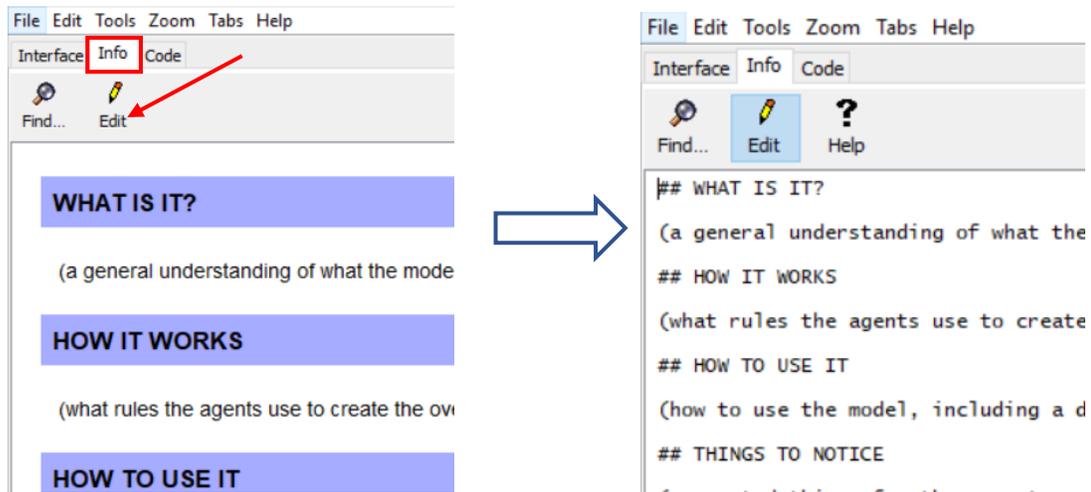


Figure 1.9, Info Tab

Code Tab

The Code tab is where the code for the model is stored.

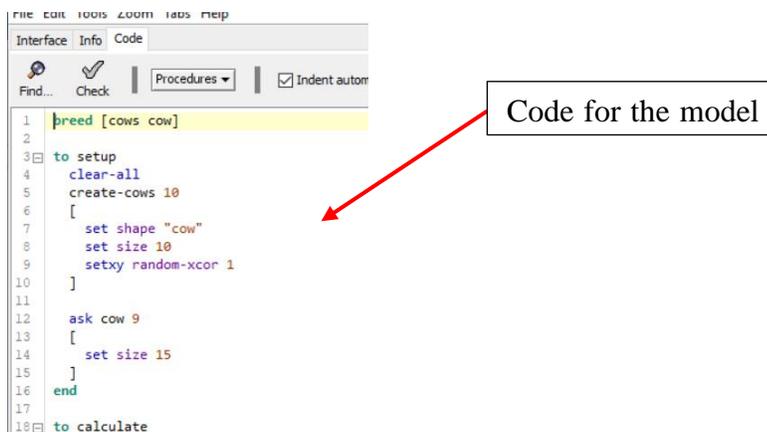


Figure 1.10, Coding Section

Icons for developing a model

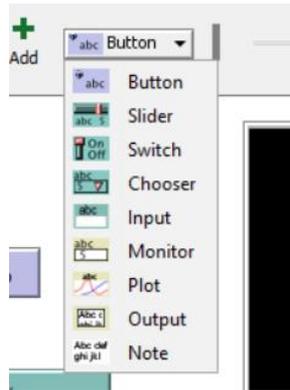


Figure 1.11, Components for developing a model

A brief explanation of the icons and functions of the icons given below.

Button: A button is either once or forever. When you click on a button once, it executes its instructions once. The forever button executes the instructions like a loop until you click on the button again to stop the action.

Slider: Sliders are global variables. They are used in models as a quick way to change a variable without having to recode the procedure every time.

Switch: Switches are a visual representation of a true/false global variable. You may set the variable to either on (true) or off (false) by flipping the switch.

Chooser: Choosers let you choose a value for a global variable from a list of choices, presented in a drop-down menu. The choices may be strings, numbers, Booleans, or lists.

Input: Input Boxes are global variables that contain strings or numbers. The model author chooses what types of values you can enter. Input boxes can be set to check the syntax of a string for commands or reporters.

Monitor: Monitors display the value of any reporter. The reporter could be a variable, a complex reporter, or a call to a reporter procedure.

Plot: Plots show data the model is generating.

Output: The output area is a scrolling area of text which can be used to create a log of activity in the model.

Note: Notes lets you add informative text labels to the Interface tab.

2. Tasks:

Task 01:

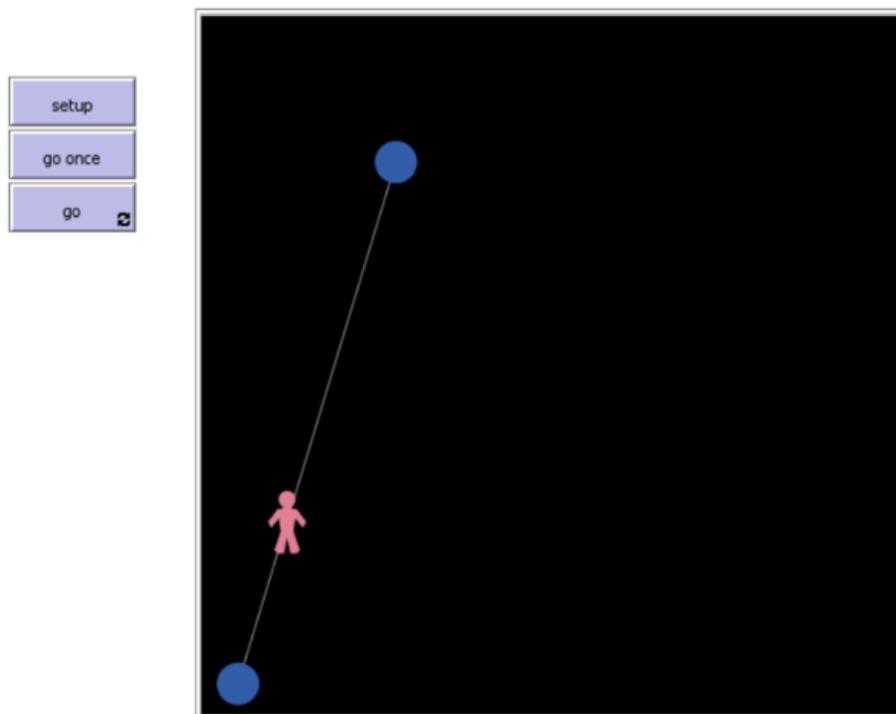
Write code to create a simple calculator with the operations “+”, “-”, “*” and “/” only.

1. There are two inputs and one operation. All these should be provided as input using the keyboard.
2. There is a button called Calculate, which should be designed so that the output appears on the monitor as soon as you click it.
3. The number of turtles should be designed using the result obtained and with the features given below.
 - Shape “person”
 - Color pink
 - Xcor and Ycor random position

Task 02:

Create two turtles with the shape of a circle. Connect them with a thin line.

1. Create a turtle with a human-shaped stay at one circle
2. Design in such a way that it passes through a line connected from one circle to another as shown below.



Task 03: Wall Avoidance

Design the blue wall in the middle of a white environment as shown below. Edit the max-pxcor and max-pycor are to 30, patch size is to 10.

- Wall should be designed between -2 and 2 (pycor).
- Create ten turtles with the shape of the butterflies in random locations.
 - Set the velocity of the butterflies is to 0.0001
 - Set the heading is to “random 360”
- All of these butterflies should be designed to fly but they should avoid that wall.

