this is the master program of the City Shaper

default sequence
route 1 - bridge
route 2 - elevator
route 3 - design and build
route 4 - crane
route 5 - ferris
route 6 - traffic gen

this program use a variable "route" to control the next route to run; the robot will wait for the enter key to start the mission (x); user can also use the Up and Down buttons to go back and forth to pick the next mission if necessary.

while running the route by myBlock, the robot also listens to the "back" button, which can interrupt the mission, this is useful if operator needs to "abort" current route and re-launch

main loop

bottom

set the variable to FALSE; we will not run route only when it is TRUE after ENTER button

by default, increase the "route" by 1 so we can run next route by simply clicking the ENTER button

display the route name

beep to indicate to operator

wait for button; note the output: 0 = nothing, 1 = center button; we can use it in the Switch block

EV3 Button ID
0 Nothing
1 Left
2 Center
3 Right
4 Up
5 Down

left button, do nothing

center bnutton to set the startRoute variable to TRUE

right button, do nothing

up button, decrease the "route" variable to move to the previous route.
Note, make sure only to do this when route is greater than 1

down button, increase the "route" variable to move to the next route (to skip current route).
Note, make sure only to do this when route is less than the maximum number

switch block to run the route based on the "route" variable.
Note : each route is one MyBlock

run route

Here we run the route MyBlock.

We couls simply just execute the MyBlock to run the route program. However, we want the ability to "abort" the route in the middle by pressing the "back" button.

For this purpose, we place the MyBlock inside a loop, and we add a parallel loop to listen to the back button. We call it "route loop" which only supposedly run once. In this loop, we have parallel tasks, one is to run the MyBlock, the other is another loop to check the Back button, let's call it "Back Button Loop"

When the operator press the Back button, the "Back Button loop" will end, and see the "Loop Interrupt" command to interrupt the "Route Loop", which will kill the running MyBlock.

However, what if operator never press the Back button? In this case, the "Back Button loop" will be stuck! To solve this problem, we need to end the "Back Button loop" when the route ends. For that, we add a variable called "routeDone" as a signal.

If the MyBlock runs until the end, it will set the variable "routeDone" to be true, and the "Back Button loop" will be over.