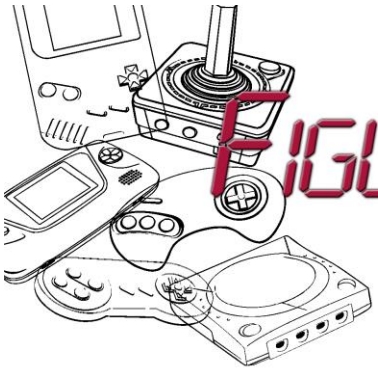
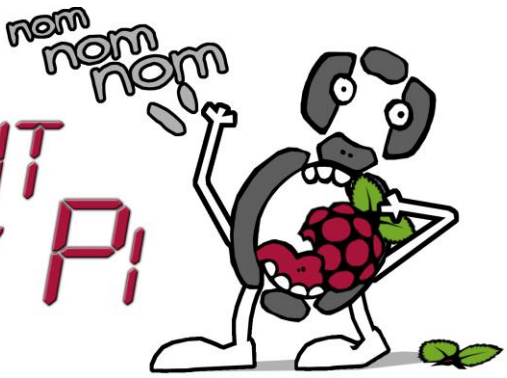


Figure Eight my Pi



The Marketing Department of ***Seven Segments of Pi Enterprises*** has created a computer games character called the ***PiSeg***

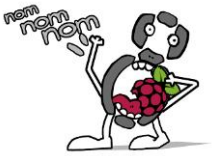
...and devised a game to run on the ***Seven Segments of Pi*** Games Console called...

Figure Eight my Pi

The cunning "PiSeg" is munching its way around your Pi...
...but when it chomps up and down the sides watch out!
There is a chance it will try and "Figure Eight Your Pi"
by attempting to gobble the prized middle segment!
If it tries to, you must shoot it!!!
If you are on target you will score a point...
...but if you let it munch past, you will lose a point!
And don't waste your ammunition!
If you shoot when the cunning "PiSeg"
is not munching the prized middle segment...
...you will lose another point!

How many can <u>you</u> score?

SEVEN SEGMENTS OF PI
FIGURE EIGHT MY PI PROJECT

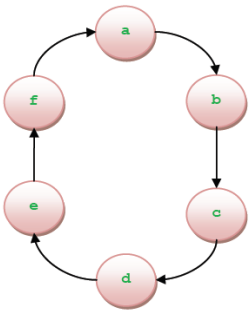


THIS IS HOW TO PLAY "FIGURE EIGHT MY PI"
ALL YOU NEED TO DO IS WRITE THE SOFTWARE!

Here are your instructions, follow them carefully

1. Start > Programming > IDLE (Opens Python Shell) and open the file FigureEightMyPi_Step0.py This is the starting software for your game. Run it to see what it does.

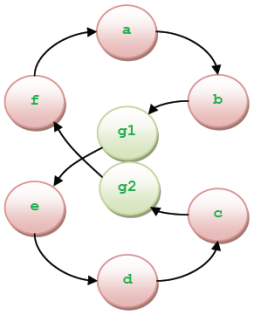
   It makes a single segment (the "*PiSeg*") appear to step around the display in a clockwise "Figure Zero" Look at the software. It is written as a 'State Machine' with 6 states

   State "a" turns segment 'a' '*on*' for 0.5 seconds. It then moves to state "b" which turns segment 'b' '*on*' for 0.5 second, then c, d, e and f then back to state "a". *Tip - Before you start each step save it with a different file name so that you could always go back a step and for each new step save as a new file name so that can always go back to your last working software!*

2. Change the code so that the *PiSeg* appears to step around the display in a "Figure Eight" and reduce the delay to 0.3 seconds to speed it up! Run it! does it work?

3. Add sound effect called 'STEP' at the end of each state just before it changes to the next state. You can use any sound effect out of those provided or use your own. Run it! You should now hear the sound the PiSeg makes as it munches round your Pi!

4. Change the state machine so that the *PiSeg* when it goes up or down the side has a 50% chance of carrying on in a "Figure Zero" and a 50% chance of going across the middle in a "Figure Eight" *Hint* – you will need to code more states since, for instance when segment 'a' is '*on*' it now might be going clockwise or anticlockwise! Run it! You should now see the PiSeg sneak across the middle …but not every time!

5. Read the PushButton at the end of each state (just before it plays the sound effect) and print "True" or "False" to the Terminal depending on whether it is being pressed. Run it! Press the PushButton and check what is printed! *Hint* – once working '*comment out*' all prints. You no longer need them and they will slow the game down!

6. Add a score. Start the score at 0. Add 1 to the score if PushButton is 'True' at the end of all 'g' segment states. Subtract 1 if PushButton is 'False' at the end of all 'g' segment states. And subtract 1 if PushButton is 'True' at the end of all other states. Print the score whenever it changes. Run it! You now almost have a working game!

7. Add a sound effects called 'HIT' whenever you add 1 to the score. Add a sound effects called 'MISS' whenever you subtract 1 from the score. Run it! You should now be able to tell if you have had a 'hit' or a 'miss'!

8. Keep a count of how many times the *PiSeg* goes through any 'g' segment state. Also reduce the 'delay' by 0.01 seconds at the end of every 'g' segment state so the *PiSeg* gradually speeds up!

9. End the game when count = 10.

10. Run it! And if it works...

...YOU HAVE NOW WRITTEN THE GAME "FIGURE EIGHT MY PI"!!!

...BUT CAN YOU MAKE IT EVEN BETTER?