

Program Code - Section 3

```
1  # Isaac Munoz
2  # Tron (2D top down view version, not fancy 3D FPV)
3  # started on 2/12/19
4  # finished by 3/1/19
5
6  # remix of "Snake Game" from: https://gist.github.com/sanchitgangwar/2158089
7  # all original code has a * next to #      (Ctrl f: #*)
8
9  '''
10  Tips for Players:
11  - Don't press the arrow key opposite your current direction
12  - Don't crash into any lines (your's or AI's)
13  - AI relies on random movement options so it's not too hard to beat
14  - Going through the borders teleports you/AI to the opposite side
15  '''
16  import curses, random, time
17  from curses import KEY_RIGHT, KEY_LEFT, KEY_UP, KEY_DOWN
18
19  # Intro
20  print "Welcome to PyTron!\nYour goal is to take out the AI by getting them to crash into your line.\nBe aware\nhowever that you also die by crashing into either your own or the AI's lines."
21
22  def start():
23      begin = str(raw_input("Are you ready?\nType y/n\n"))
24      if begin == 'y':
25          game()
26      elif begin == 'n':
27          print "You're no fun."
28      else:
29          print "That's not an option." # you can still manually put "start()" or "game()"
30
31  def game():
32      # initialize window (length = 58) (height = 18)
33      score = 0
34
35      curses.initscr() #*(to line 41) allows use of curses
36      win = curses.newwin(20, 60, 0, 0) # window is 2 more x & y b/c of border characters
37      win.keypad(1)
38      curses.noecho()
39      curses.curs_set(0)
40      win.border(10)
41      win.nodelay(1)
42
43      key = KEY_UP #* begin moving
44      ckey = KEY_DOWN
45      Player = [[1,19], [4,9], [4,8]] #* Player start co-ordinates
```

```

46 AI = [[1,38], [4,9], [4,8]] # AI start co-ordinates
47
48 while key != 27: #*(to line 76) Esc = end game
49     win.border(0)
50     win.addstr(0, 27, ' PyTron ') # just the title at the top of the border
51
52     win.timeout(150 - (len(Player)/5 + len(Player)/10)%120) # Increases the speed of Player as its length
    increases
53     win.timeout(150 - (len(AI)/5 + len(AI)/10)%120) # Increases the speed of AI as its length increases
54
55     prevKey = key # Previous key pressed
56     pause = win.getch()
57     key = key if pause == -1 else pause
58
59     if key == ord(' '): # Space Bar = pause/resume
60         key = -1
61         while key != ord(' '):
62             key = win.getch()
63         key = prevKey
64         continue
65
66     if key not in [KEY_LEFT, KEY_RIGHT, KEY_UP, KEY_DOWN, 27]: # only arrow keys (& Esc) work
67         key = prevKey
68
69 # Increase Player length in current direction
70 Player.insert(0, [Player[0][0] + (key == KEY_DOWN and 1) + (key == KEY_UP and -1), Player[0][1] + (key
    == KEY_LEFT and -1) + (key == KEY_RIGHT and 1)])
71
72 # Player teleports to opposite side if hits border
73 if Player[0][0] == 0: Player[0][0] = 18
74 if Player[0][1] == 0: Player[0][1] = 58
75 if Player[0][0] == 19: Player[0][0] = 1
76 if Player[0][1] == 59: Player[0][1] = 1
77 # AI teleports to opposite side if hits border
78 if AI[0][0] == 0: AI[0][0] = 18
79 if AI[0][1] == 0: AI[0][1] = 58
80 if AI[0][0] == 19: AI[0][0] = 1
81 if AI[0][1] == 59: AI[0][1] = 1
82
83 #* If Player runs over line
84 if Player[0] in Player[1:]: break
85 if Player[0] in AI[1:]: break
86 # if AI runs over line
87 if AI[0] in Player[1:]: break
88 # AI is allowed to go through its own line
89 # Uncomment line 89 to enable, but I found that this happens too quickly into the game to really play

```

```

89     # Uncomment line 89 to enable, but I found that this happens too quickly into the game to really play
90     # if AI[0] in AI[1:]: break
91
92     win.addch(Player[0][0], Player[0][1], '#')          #* Player
93     win.addch(AI[0][0], AI[0][1], '*')                  # AI
94
95     prevkey = ckey
96     moves = [KEY_DOWN, KEY_UP, KEY_RIGHT, KEY_LEFT]      # AI moveset
97     if ckey == prevkey:                                   # AI can't go back into line
98         time.sleep(0.01) # manual delay (so it doesn't go too fast; higher = slower)
99         ckey = random.choice(moves)
100         if (prevkey == KEY_DOWN) and (ckey == KEY_UP):    # no up if going down
101             ckey = random.choice(moves)
102         if (prevkey == KEY_UP) and (ckey == KEY_DOWN):    # no down if going up
103             ckey = random.choice(moves)
104         if (prevkey == KEY_LEFT) and (ckey == KEY_RIGHT): # no right if going left
105             ckey = random.choice(moves)
106         if (prevkey == KEY_RIGHT) and (ckey == KEY_LEFT): # no left if going right
107             ckey = random.choice(moves)
108         # just stops AI from moving in reverse
109
110     # increase AI length
111     AI.insert(0, [AI[0][0] + (ckey == KEY_DOWN and 1) + (ckey == KEY_UP and -1), AI[0][1] + (ckey ==
112     KEY_LEFT and -1) + (ckey == KEY_RIGHT and 1)])
113
114     score += 0.1
115
116     curses.endwin() #*
117
118     print "\nGame over."
119     print "You survived for", (score + 2), "seconds."
120     play = str(raw_input("Would you like to play again?\nType y/n\n"))
121     if play == 'y':
122         game()
123     else:
124         print "Ok."
125
126 start()

```