The MagPi
ESSENTIALS

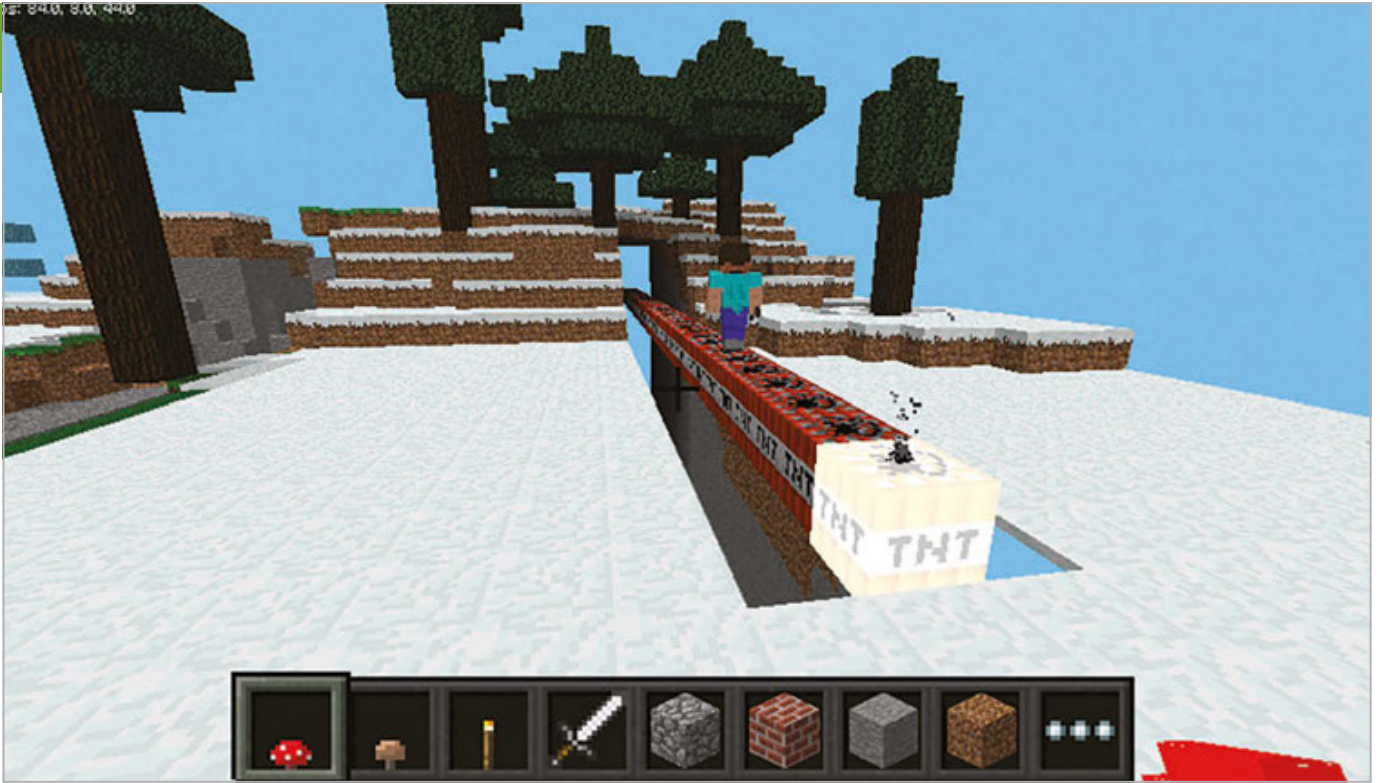# [CHAPTER FIVE]
TNT RUN!

Can you outrun an explosion? Test your speed with this marvellous Minecraft mini-game!

One of the many amazing things about Raspberry Pis is that they have their own edition of Minecraft for free; what's even better is you can code it in Python using the Minecraft API! In the next 30 minutes you'll create a game called TNT Run, in which you start at one end of a long line of TNT and have to make it to the safe area without the TNT exploding in your face.

The Minecraft API gives us complete control over many elements of the game; this includes teleporting players around the world and displaying helpful messages on the screen. We are also able to place blocks automatically: not just one at a time, but as a three-dimensional collection of blocks.

This game will also include a block which is unique to Pi and Pocket editions: invisible bedrock. We use this block to keep the TNT from falling to the ground when lit, and as an invisible path leading to the safe area. You may find something strange about this block when you place it next to a non-invisible block, so try experimenting with that; you need to look directly into the invisible bedrock.

TNT also behaves differently in the Pi edition. Whereas in other editions you set off TNT with flint and steel, a fire charge or a flaming arrow, in the Pi edition you just need to hit it a couple of times with

anything. However, you can't just do this with any old TNT block: first, we need to set its block data value to an odd number (1, 3, 5, 7, or 9). Most blocks have a block data value and by changing this, it will alter the block's behaviour. For example, when you set Nether Reactor Core's block data value to 1, it appears in a red colour; if you set it to 2 then it will come out as a dark blue colour. We use these cool-looking blocks to mark where the teleporter is and as a part of the end podium. The teleporter function in the code is designed so that if you manage to get to a certain point along the line of TNT, it gives you a boost and teleports you forward.

When you create your Minecraft world, fly around it and find a cool location to create your TNT course (i.e. not when you are near a cliff or the end of the world). Your code takes the player's position and constructs the TNT course, using this location as the starting point. When you run the code, make sure Minecraft is running, otherwise your code will give you a connection error.

Once you have finished the project, you can customise your own version of the TNT Run game by adding changes like making the row of TNT longer, or creating a fancier safe area. Also, when you're connected to a network with other Raspberry Pis, you can join someone else's world, create two lines of TNT and race to the safe areas simultaneously!
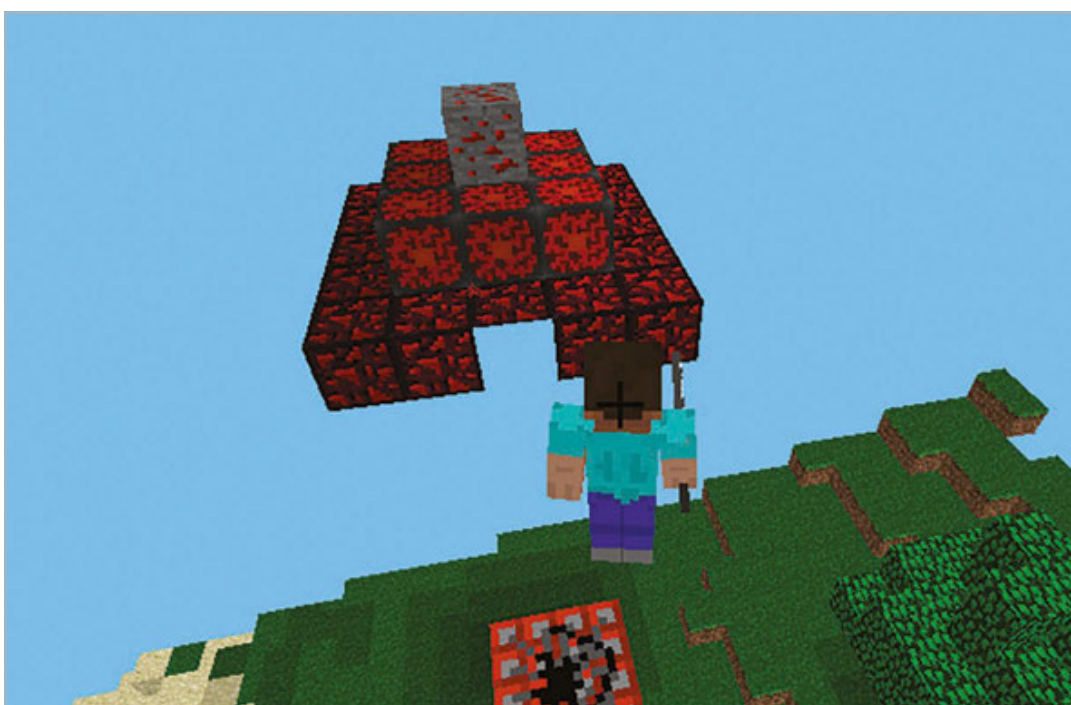
Left This is the safe area you need to run to: if you make it there, you've won

TNTRUN.py

```
01.  # import all the necessary modules
02.  from mcpi.minecraft import Minecraft
03.  from mcpi import block
04.  import time
05.
06.  # connect with the Minecraft world
07.  mc=Minecraft.create()
08.
09.  # get the player's position
10.  pos=mc.player.getTilePos()
11.
12.  # check if the end of the world will engulf your creation and move you if you're too close
13.  if pos.z<-40:
14.      mc.postToChat(
15.      'teleporting to safer distance in progress!')
16.      mc.player.setPos(pos.x,pos.y,-40)
17.      pos=mc.player.getTilePos()
18.
19.  # mark where the teleport is
20.  zpos=pos.z-40
21.
22.  # create the valley by hollowing it out with air
23.  mc.setBlocks(pos.x-1,pos.y+3,pos.z,pos.x+1,pos.y-7,pos.z-88,block.AIR.id)
24.
25.  # build the invisible bedrock support
26.  mc.setBlocks(pos.x,pos.y-1,pos.z,pos.x,
27.  pos.y-7,pos.z,block.BEDROCK_INVISIBLE.id)
28.  mc.setBlocks(pos.x-1,pos.y-1,pos.z,pos.x,
29.  pos.y-7,pos.z,block.BEDROCK_INVISIBLE.id)
30.  mc.setBlocks(pos.x+1,pos.y-1,pos.z,pos.x,
31.  pos.y-7,pos.z,block.BEDROCK_INVISIBLE.id)
32.  mc.setBlocks(pos.x,pos.y-1,pos.z-88,pos.x-1,pos.y-7,pos.z-88,block.BEDROCK_INVISIBLE.id)
33.  mc.setBlocks(pos.x-1,pos.y-1,
34.  pos.z-88,pos.x,pos.y-7,pos.z-88,
35.  block.BEDROCK_INVISIBLE.id)
36.  mc.setBlocks(pos.x+1,
```

```
37.   pos.y-1,pos.z-88,pos.x,
38.   pos.y-7,pos.z-88,block.BEDROCK_INVISIBLE.id)
39.   mc.setBlocks(pos.x,pos.y,pos.z,pos.x,
40.   pos.y-7,pos.z-92,block.BEDROCK_INVISIBLE.id)
41.
42.   # build the bomb
43.   mc.setBlocks(pos.x,pos.y,pos.z,pos.x,pos.y,
44.   pos.z-88,block.TNT.id,1)
45.
46.   # build the end podium
47.   mc.setBlocks(pos.x-2,pos.y,pos.z-93,
48.   pos.x+2,pos.y,pos.z-97,block.GLOWING_OBSIDIAN.id)
49.   mc.setBlocks(pos.x-1,pos.y+1,pos.z-94,pos.x+1,
50.   pos.y+1,pos.z-96,block.NETHER_REACTOR_CORE.id,1)
51.   mc.setBlock(pos.x,pos.y+2,pos.z-95,
52.   block.REDSTONE_ORE.id)
53.
54.   # set how many teleports you have
55.   teleport=1
56.
57.   # build the display teleport signal block
58.   mc.setBlock(pos.x+1,pos.y+1,pos.z-44,
59.   block.NETHER_REACTOR_CORE.id,2)
60.   mc.setBlock(pos.x-1,pos.y+1,pos.z-44,
61.   block.NETHER_REACTOR_CORE.id,2)
62.
63.
64.   # teleport player when at a certain position
65.   while teleport ==1:
66.       pos=mc.player.getTilePos()
67.       if pos.z==zpos:
68.           mc.player.setPos(pos.x,pos.y,pos.z-24)
69.           teleport=0
```