Hackpack Anthology

Contents

3

Computer Science Skills competencies checklist

Based on NC and @Craigarghs brilliant Python programming with Minecraft book checklist(Reproduced with his kind permission :))

Syntax

Variables		Changing Variables	
Data types		Statements	
Integers		Whitespace and Tabs	
		Single-lIne comments	
Floats		Multi-line Comments	
Boolean			

Maths Operations

Expressions and statements		Exponentials	
Maths operators		Modulo	
Addition		Operator order	
Subtraction		Interchanging variables and values	
Multiplication		Short hand operators	
Division			

String and console output

Strings		Placeholders	
Substrings		Console Input	
String functions -len() -lower() -upper() -str()		Date and Time	
Print			
Concatenation			

Comparators and Control Flow

Comparators including:		Greater than or equal to (>=)	
-Equal to (==)		Boolean Operators	
-Not equal to (!=)		If statements	
-Less than(<)		Else statements	
-Less than or equal to (<=)		Elif statements	
-Greater than (>)			

Functions

Creating and Calling Functions		Built in functions including:	
Returning a value		-max	
Arguments		-min	
Modules		-abs	
Importing modules		-type	

Lists and Dictionaries

Creating lists		Removing items	
Accessing index positions		For loop	
Adding items		Sorting a list	
List length		Combining lists	
Slicing		Defining a dictionary	
Searching a list		Changing / adding items in a dictionary	
Inserting an item		Deleting items in a dictionary	

Functions and Lists

Lists as arguments		Splitting a string into a list	
Modifying ever list item		Multi-dimension lists	
Range Function		Joining two lists	
Converting a list into a string		Undefined number of lists	

Loops

While Loops		Strings as lists	
Boolean Operators with While Loops		Looping dictionaries	
Infinite Loops		Indexes and for loops	
Break		Zipping two lists	
While/else		For /else loops	
For Loops		For / else break	

File Input and Output

Opening a file		Reading a line	
Writing and closing a file		Automatically closing a file	
Reading a file		Closed attribute	

Classes and Object Orientated Programming

Creating classes		Creating objects	
-init-()		Accessing attributes	
Arguments		Class scope	

Creating methods		Overiding methods and attributes	
Multiple objects		Referencing superclass methods in a subclass	
Inheritance			

Programmer competencies

Reusing code		Working in team	
Decomposing a problem		Sharing knowledge	
Problem solving		Testing	
Persevering when a program doesn't work		Peer review and constructive feedback	
Systems thinking understanding how parts of a program relate to each other		Chosing the correct technology for the solution.	
Communication with others		Requirements analysis (understanding and prioritising the different needs of the system)	

MINECRAFT COMPETITIONS MODDING CREATIVE

Hack pack Volume 1
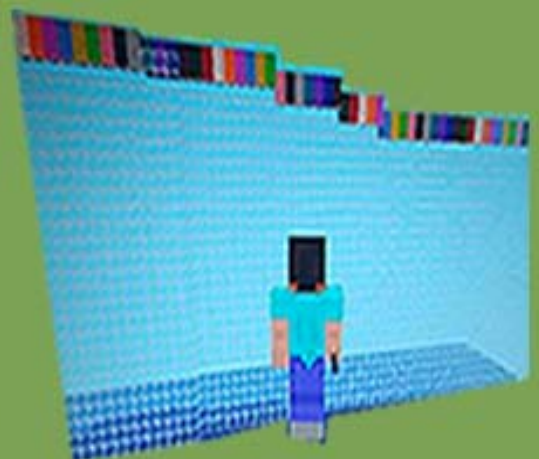
Volume 1

Hello and welcome to the first edition of the Minecraft Comps hack pack. This has been created by teachers who use / experiment with Minecraft Pi at school to help kids learn about computing / computational thinking. Included in this first edition are worksheets / ideas that you could use to help introduce Pi at school or as part of a jam / lunch club after school club. We hope you find these ideas helpful and make coding with Minecraft Pi a bit less daunting. Contributors include: Chris Penn, Dan Aldred and Ben Davies.

Hide a Diamond, Find a Diamond (Dan Aldred)

This hack hides a diamond in the Minecraft world, it then uses the 'set camera' code to show you a bird's eye view of the location of the diamond. Then it is your quest to seek out and find the treasure.

```python
from mcpi import minecraft

from time import sleep

mc = minecraft.Minecraft.create()

mc.postToChat("Here is the diamond block I have hidden.")

mc.setBlock(100, 25, 100, 57)

mc.camera.setFixed()

mc.camera.setPos(100, 30, 100)

sleep(10)

mc.postToChat("Go find it!")
mc.camera.setNormal()
```

<u>Questions</u> to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

The Midas Touch (Dan Aldred)

Be careful what you wish for, everything you touch turns to gold.

```
from mcpi import minecraft
Import time
mc = minecraft.Minecraft.create()


gold = 41

water = 9

air = 0

while True:

  pos = mc.player.getTilePos()

  blockBelow = mc.getBlock(pos.x, pos.y - 1, pos.z)


  if blockBelow != air and blockBelow != water:

            mc.setBlock(pos.x, pos.y -1, pos.z, gold)

  time.sleep(0.1)
```

<u>Questions</u> to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

The Sand Drop (Dan Aldred)

Watch out, in the Minecraft World there are some strange goings on, random blocks of sand falling from the sky, watch out they don't hit you or it's game over!.

```
from mcpi import minecraft
from time import sleep
mc = minecraft.Minecraft.create()


pos = mc.player.getTilePos()

while mc.getBlock(pos.x, pos.y, pos.z) != 13:

    mc.setBlock(pos.x, pos.y + 25, pos.z, 13)

    sleep(1)

    pos = mc.player.getTilePos()

mc.postToChat("Got you!")
```

<u>Questions</u> to answer about what you have coded and reflect on what you have learned:

Code concepts used?

19

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend </u>this script?

This modding challenge is built on from the original walking on Ice modd by Craig Richardson, who in turn may have got it from elsewhere (I have no idea). In the original if you are walking on water then the block is then turned into ice. I have adapted this to follow the following logic:

Generate a list of all block types ids

While true loop / do stuff

  Wait a short time

      Lay a block of ice directly beneath Steve

      Place another random block directly beneath the Ice block (i.e. 2 below Steve)

      If block 2 beneath Steve is tnt

        Activate the tnt block

        Change block 1 below Steve from ice to lava

Here is the code I created for this basic mod:

```python
from mcpi import minecraft
import mcpi.block as block
import time
import random
mc = minecraft.Minecraft.create()

#generate a list of all known block types ids for mcpi
blockTypeList= [46,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,15,16,17,18,20,21,22,24

while True:#do stuff
    time.sleep(0.4)
    pos = mc.player.getPos()
    x = pos.x
    y = pos.y
    z = pos.z

    mc.setBlock(x,y-1,z,79)#place down an ice block directly underneath steve
    mc.setBlock(x,y-2,z,random.choice(blockTypeList))#place a random block under

    #check the block two below steve
    blockTwoBelow = mc.getBlock(x,y-2,z)

    #if it is tnt then activate it
    if blockTwoBelow == 46:#activate tnt
        mc.setBlock(x,y-2,z,46,1)
        mc.setBlock(x,y-1,z,10)#Lava = very small boom, I think
```

Here is a screen grab of it in action:

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you extend this script?

What happens if you make changes?

Code in python 2.7 (edited and hacked code of Ben Davies for Minecraft comps hack pack May 2015 edition)

Code

```python
import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time




def SetDiscoFloor(index):
    pos = mc.player.getPos()
    mc.setBlocks(pos.x, pos.y-1, pos.z,
    pos.x + 35, pos.y-1, pos.z + 35,block.WOOL.id,index)#changes the colour of w

def SetDiscoFloorTNT():
     mc.setBlocks(pos.x-3, pos.x-2, pos.x-1, pos.x, pos.x+1, pos.x+2,46)

mc.postToChat("Watch Out for disco blocks")

Count = 0

while True:

    SetDiscoFloor(Count)
    time.sleep(0.2)
    if Count == 25:
        SetDiscoFloorTNT()
    if Count == 35:
        Count = 0
    Count = Count +1
```

 This is what it creates the following:
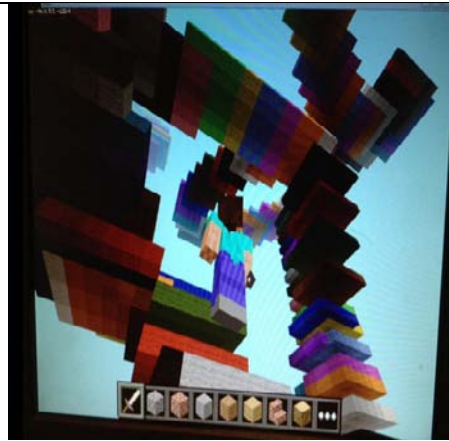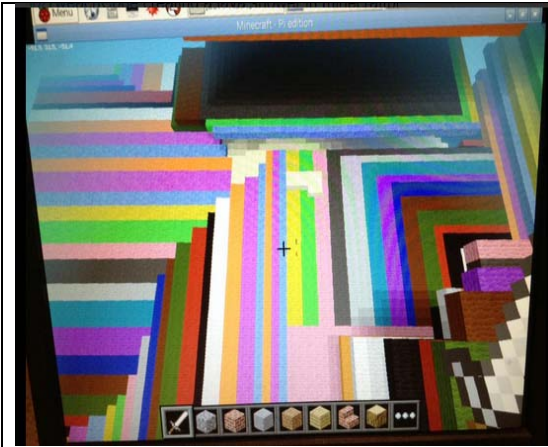
24

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

25

How could you <u>extend</u> this script?

What happens if you make changes?

Visualising/graphing live data with Astro Pi/ sense hat and Minecraft Pi (@ncscomputing)

Code

```python
from astro_pi import AstroPi
import time
import mcpi.minecraft as minecraft
import mcpi.block as block
import random

ap = AstroPi()

mc = minecraft.Minecraft.create()

mc.postToChat("Start Graph")


TempList=[]

DataStreamCount=0
woolCount = 0

def setGraphLine(temperature,woolCount):
    pos = mc.player.getPos()
    x,y,z = mc.player.getPos()
    mc.player.setPos(pos.x+3,y,z)
    mc.setBlocks(pos.x,pos.y,pos.z,pos.x,pos.y+int(temperature-1),pos.z,0)#set air
    mc.setBlocks(pos.x,pos.y+int(temperature-1),pos.z,pos.x,pos.y+int(temperature),pos.z,35,woolCount)#set tip of graph

pos = mc.player.getPos()

while True:
    temperature = ap.get_temperature()
    print("temp",int(temperature))
    time.sleep(2)
    TempList.append(temperature)
    print (TempList[DataStreamCount])
    setGraphLine(temperature,woolCount)
    woolCount = woolCount+1

    if woolCount == 16:#reset wool count when max is reached
        woolCount = 0
```

This is what it creates the following:
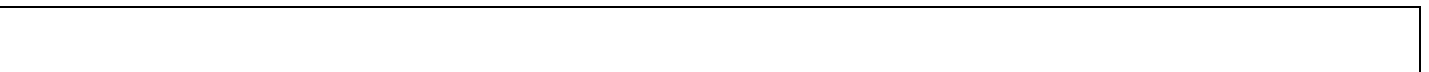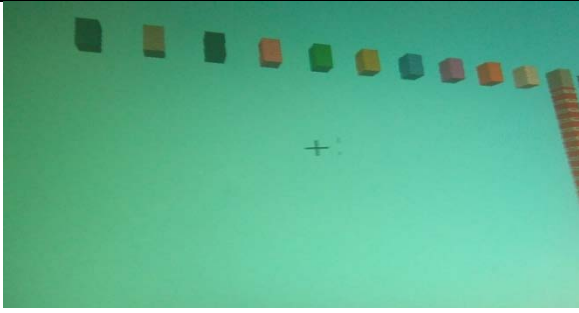
Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you extend this script?

What happens if you make changes?

Code

```python
import mcpi.minecraft as minecraft
import mcpi.block as block


mc = minecraft.Minecraft.create()

mc.postToChat("Welcome to my Interactive world:)")


while True:

    evs = mc.events.pollBlockHits()
    for e in evs:
        pos = e.pos


        b=mc.getBlock(pos.x,pos.y,pos.z)

        if b == 46:
            mc.postToChat("Tnt")
        elif b == 20:
            mc.postToChat("Glass")
        elif b == 45:
            mc.postToChat("Brick")
        elif b == 26:
            mc.postToChat("Bed")
        elif b == 2:
            mc.postToChat("Grass")
        else:
            mc.postToChat("Other")
```

This is what it creates the following:
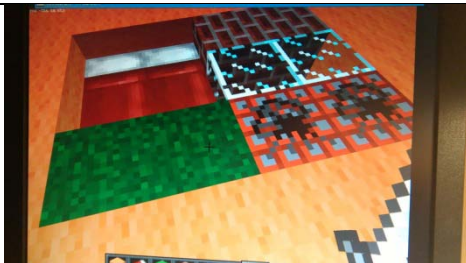
Right click and whack the block/ bed and messages appear in the chat window

30

<u>Questions to answer about what you have coded and reflect on what you have learned:</u>

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

What happens if you make changes?

Exploding Wall (@b3ndavi3s)

In this activity you will write a program that creates a TNT wall that you can detonate with your sword.

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and select Start Game	Open Python 2 (menu - programming - python 2)
- get the player's position - create a cuboid of active TNT - post messages - create loops - use delays to sequence the events	Select an existing world or Create a New World Use the tab key to release the cursor from minecraft	Open a new file (file - new) Save the new file as exploding wall (file - save as)

What are you trying to do?	Your pseudo-code

```import mcpi.minecraft as minecraft``` ```import mcpi.block as block``` ```mc = minecraft.Minecraft.create()``` ```mc.postToChat("exploding wall")```
 	
 ```pos = mc.player.getPos()``` ```mc.setBlocks(pos.x-3, pos.y-1, pos.z+3,``` ```   pos.x+3, pos.y+3, pos.z+5,``` ```   block.TNT.id,1)``` ```mc.postToChat("Hit a block & stand back")```
 	
 ```#now press ctrl+s to save your program then F5 to run it.```
	Import modules from minecraft and python to use in your program
	Connects the python program to the minecraft game
	Print the message inside "" to the screen
	
	Gets the player's current position as the coordinates
	Sets the coordinates for the cuboid
	
	Sets the block to be used (TNT.id,1 is exploding TNT)

Challenges
- replace the the block with TNT.id and see what happens
- increase the size of the wall by changing the coordinates. The larger the wall, the longer it will take for the graphics to render

```
            mc.setBlocks(pos.x-3, pos.y-1, pos.z+3,
                    pos.x+3, pos.y+3, pos.z+5,
                        block.TNT.id,1)
```

Lava Shower (@b3ndavi3s)

In this activity you will write a program that creates a lava shower that starts above you in the sky.

34

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and select Start Game	Open Python 2 (menu - programming - python 2)
get the player's position **create a cuboid of flowing lava** **post messages** **create loops** **use delays to sequence the events**	**Select an existing world or Create a New World** **Use the tab key to release the cursor from minecraft**	**Open a new file (file - new)** **Save the new file as lava shower (file - save as)**

What are you trying to do?	Your pseudo-code

In your new file type the following code

```import mcpi.minecraft as minecraft``` ```import mcpi.block as block``` ```import time``` ```mc = minecraft.Minecraft.create()``` ```while True:``` ``` mc.postToChat("Look up!")```	**Import modules from minecraft and python to use in your program** **Connects the python program to the minecraft game** **Creates a forever loop**

35

| ```
pos = mc.player.getPos()
mc.setBlocks(pos.x-5, pos.y+10, pos.z-5,
            pos.x+3, pos.y+15, pos.z+5,
            block.LAVA_FLOWING.id)
mc.postToChat("Lava Shower!")
time.sleep(2)
x=0
while x<5:
    x=x+1
    mc.postToChat("RUN!!!")
    time.sleep(1)
mc.postToChat("Phew!")
time.sleep(5)
``` | Gets the player's current position as coordinates<br>Gives the coordinates for the lava<br><br>Sets the blocks to flowing lava<br><br><br><br><br>Creates a variable called x with a value of 0<br>Creates a loop for 5 occasions<br>Increase the variable x by 1<br>The double indent showing it's a loop within a loops |
| #now press ctrl+s to save your program then F5 to run it. | |

Challenge

Can you change the type of shower?

Can you alter the amount of time before another lava shower starts?

Can you make the lava shower higher/lower?

Can you change the number of times the message run appears on the screen?

Explain how you modified the code

Lava Shower with triggers (@b3ndavi3s)

In this activity you will modify your lava shower to start if certain blocks are stood on

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and select Start Game	Open Python 2 (menu - programming - python 2)
- check the block the player is standing on - create if and else statements	Select an existing world or Create a New World Use the tab key to release the cursor from minecraft	Open a recent file (lava shower) Save the new file as lava shower trigger (file - save as)

What are you trying to do?	Your pseudo-code

Use a trigger (selection statement) to start the lava shower.
In the example below the lava shower starts when the character stands on block of sand.
Add the bold lines to your program

```
import mcpi.minecraft as minecraft
import mcpi.block as block
import time
mc = minecraft.Minecraft.create()
while True:
     pos = mc.player.getPos()
    blockBelow = mc.getBlock (pos.x, pos.y-1, pos.z+1)
    if blockBelow == block.SAND.id
       mc.postToChat("Look up!")
       mc.setBlocks(pos.x-5, pos.y+10, pos.z-5,
                    pos.x+3, pos.y+15, pos.z+5,
                    block.LAVA_FLOWING.id)
       mc.postToChat("Lava Shower!")
       time.sleep(2)
       x=0
       while x<5:
            x=x+1
            mc.postToChat("RUN!!!")
            time.sleep(1)
       mc.postToChat("Phew!")
       time.sleep(5)

#now press ctrl+s to save your program then F5 to
run it.
```

check the block you are standing on and if
that blocks is sand then start the lava
shower sequence

Gets the block the player is standing on
Conditional statement - the program
below will only run if this condition is met
(stands on a block of sand)

== is used for equals, as = sets variables

Challenge
Can you change the trigger to another block?
Can you make the trigger a block above or in front of the character?

Explain how you modified the code

What Lies Beneath (@b3ndavi3s)

In this activity you will write a game in which you have to avoid the randomly selected block of air

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and select Start Game	Open Python 2 (menu - programming - python 2)
- get the player's position - create a surface area for the game - create lists - use random selection - create loops - use delays - use selection statements	Select an existing world or Create a New World Use the tab key to release the cursor from minecraft	Open a new file (file - new) Save the new file as what lies beneath (file - save as)

What are you trying to do?	Your pseudo-code

Challenges
- **change the scoring system**
- **remove more blocks**
- **make a shorter countdown timer**

| ```
import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time
mc = minecraft.Minecraft.create()
pos = mc.player.getPos()
posxlist= (pos.x-3, pos.x-2, pos.x-1, pos.x, pos.x+1,
pos.x+2)
poszlist= (pos.z+3, pos.z+4, pos.z+5, pos.z+6, pos.z+7,
pos.z+8)
mc.setBlocks(pos.x - 3, pos.y-1, pos.z +3,
          pos.x + 2, pos.y -1, pos.z+8,
          block.WOOL.id,3)
time.sleep(2)
mc.postToChat("Pick a Block")
n = 0
life = 1
while life == 1:
      x = 10
      while x>0:
            mc.postToChat(x)
            x = x-1
            time.sleep(1)
      mc.postToChat("Watch Out")
      randxPos = random.choice(posxlist)
      randzPos = random.choice(poszlist)
      mc.setBlocks(randxPos, pos.y-1, randzPos,
                randxPos+1, pos.y-500, randzPos+1,
                block.AIR.id)
      pos = mc.player.getPos()
      blockBelow = mc.getBlock(pos.x, pos.y - 1, pos.z
+ 1)
      if blockBelow == block.AIR.id:
            mc.postToChat("NO!!! Game Over")
            time.sleep(1)
``` | Import modules from minecraft and python to use in your program

Connects the python program to the minecraft game

Gets your current position

Creates lists of x and z values

Creates a cuboid of wool

Creates variable n (this will be your score)

Creates variable life

If you are alive run the following code

creates a countdown timer

creates a random column of air use the lists created earlier |

`        mc.postToChat("You scored")`
`        mc.postToChat(n)`
`        life = 0`
`    else:`
`        mc.postToChat("Phew. Now pick another block")`
`        n = n + 100`
`#now press ctrl+s to save your program then F5 to run it.`
	Checks to see if you are now standing on a block of air:
	if you are game over and posts your score; if not you get 100 points and play again.

@crazysqeak's python buildings link

https://goo.gl/Kj7pJ2

TNT Excavation (@ncscomputing)

HACKCARD 5
TNT EXECAVATION

Code:

This is the most popular hack that I have seen. Lay one strike it and move Steve around the general area. You will quickly blow massive holes in the ground. The block id for TNT is 46, the extra 1 in setBlock activates the TNT.

```
File  Edit  Format  Run  Options  Windows  H

import mcpi.minecraft as minecraft
import mcpi.block as block
mc = minecraft.Minecraft.create()

while True:
    pos = mc.player.getPos()
    mc.setBlock(pos.x,pos.y,pos.z,46,1)
    print "Boom"
```

Output in game(if you leave it long enough):

HACKCARD 5
TNT EXECAVATION

Code:

Possible extensions:

Change the block id's

Block ids can be accessed here from Raspberry Pi spy's website

http://goo.gl/FN45Ev

Add a time delay

Try water, what happens?

Try lava, what happens?

Key questions to think about :

Questions to answer about what you have coded and reflect on what you have learned:

1. Code concepts covered?

2. What role do they perform in the script?

3. Errors encountered and solutions?

4. What happens if you make changes?

Random Colour Wall (@ncscomputing)

Code:

```
File Edit Format Run Options Windows Help

import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time


mc = minecraft.Minecraft.create()



Block = 35
WoolIdList=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16]

myList = [[0 for j in range(20)]for i in range(20)]

def PrintWall():

    pos = mc.player.getTilePos()
    mc.player.setPos(pos.x,pos.y,pos.z)

    for row in range (0,20):
        for column in range (0,20):
            mc.setBlock(pos.x+row,pos.y+column,pos.z-20,Block,random.choice(WoolIdList))

while True:

    PrintWall()
    time.sleep(0.5)
```

Output in game:

MINECRAFT COMPETITIONS
MODDING
CREATIVE
Hackpack V1: Mini Hacks

HACKCARD 4
RANDOM COLOUR WALL

Code:

Possible extensions:

Change the block ids used

Change the size of the wall

Create a variable for the size of the wall

Change the time delay

Use list of block types instead of one block type

Block ids can be accessed here from Raspberry Pi spy's website

http://goo.gl/FN45Ev

Key questions to think about :

Questions to answer about what you have coded and reflect on what you have learned:

1. Code concepts covered?

2. What role do they perform in the script?

3. Errors encountered and solutions?

4. What happens if you make changes?

MINECRAFT COMPETITIONS
MODDING
CREATIVE
Hackpack V1: Mini Hacks

MINI HACKS: PROTOTYPE PIXELATED ART GALLERY

Instructions:

Steps:
Stage 1: Unlocking the folders
1. Click on menu > accessories > file manager
2. Type in the following folder location in /opt/minecraft-pi/data/images
3. Press enter
4. Press f4 and then type in the following in LX terminal(the black screen) sudo chmod 777 mob
5. Press enter
6. Press f4 and then type in the following in LX terminal(the black screen) sudo chmod 777 art
7. Press enter
Stage 2: create the skin
8. Use websites like http://www.minecraftskins.com/ to create a skin of your "artist"
9. Save a copy of it and call it 'char.png'
10.   Delete the existing char1.png
11.   Paste this file in the following folder /opt/minecraft-pi/data/images/mob
Stage 3: Create the pictures
12.   Make a copy of the file kz in the /opt/minecraft-pi/data/images/art folder
13.   Paste it onto a memory stick
14.   Switch to the PC and open it up in photo editing software. We have fireworks
15.   Paste in images over the top of existing images

16.   Save this image back again onto the memory stick
17.   Put the memory stick back into the pi. Save the new kz file in the /opt/minecraft-pi/data/images/art
Stage 3: Build the gallery in Minecraft Pi
18.   Build a big building in minecraft that has plenty of windows and big wall space. A big warehouse structure.
19.   Add lots of pictures and you should see your images appear.
20.   Pixelated gallery complete.

50 ways to say hello V1 (@ncscomputing)

Code:

```python
import mcpi.minecraft as minecraft
import mcpi.block as block
import time

mc = minecraft.Minecraft.create()

#print to shell
print "Hello :)"

#enough time to see the next bit
time.sleep(2)

#print to chat window
mc.postToChat("Hello :)")

#using a variable
msg = "Hello"
print msg
#enough time to see the next bit
time.sleep(2)
mc.postToChat(msg)


#using a mixture of variables
name = "Steve"
print msg,"",name
#enough time to see the next bit
time.sleep(2)
fullMsg = msg,"",name
mc.postToChat(fullMsg)
```

Output (Shell):

```
>>> ====================
>>>
Hello :)
Hello
Hello   Steve
>>> |
```

Output (In game):

MINECRAFT COMPETITIONS MODDING CREATIVE
Hackpack V1: Mini Hacks

MINI HACKS: PROTOTYPE 50 WAYS TO SAY HELLO!

Key questions to think about :

Questions to answer about what you have coded and reflect on what you have learned:

1. Code concepts covered?

2. What role do they perform in the script?

3. Errors encountered and solutions?

4. What happens if you make changes?

Possible extensions:

1. Add a while loop

2. Add a for loop

3. Use raw_input()

Volume 2

#Hackpack Volume 2.1
Alpha version

It seems apt to say a big thank you to the following people for their contributions thus far:

- Dan Aldred for the consistent input, help and great programming ideas

- Ben Davies, Rob Jones-Cowley, Elton Lane for their great hacking contributions.

- Crazysqueak for proving that age is no barrier to be able to inspire others

- David Whale, Martin O'Hanlon for their continued advice and great book with excellent ideas

- Craig Richardson for his brilliant book

- Pipsta Printers for allowing us to have a pipsta printer to provide even more hacks going forward

- Raspberry Pi for inventing and providing a sense hat.

- Nicholas Chamberlaine School for providing the resources and support to host Raspberry Jams allow all this to happen ☺

Done, Now onwards. Let's hack ☺

Welcome

Hello and welcome to the hack pack volume 2. This has been created by teachers who use / experiment with Minecraft Pi at school to help students learn about computing / computational thinking.

Included in this 2nd edition are worksheets / ideas that you could use to help introduce Pi at school or as part of a jam / lunch club after school club. We hope you find these ideas helpful and make coding with Minecraft Pi a bit less daunting.

Contributors include: Chris Penn, Dan Aldred and Ben Davies, Elton Lane, Rob Jones-Cowley, Crazy sqeak and Sarah Zaman. We have included more complexity and variety in volume. Furthermore we have now evolved into Sonic Pi which is using coding to create music with a visual output possible via Minecraft(Its awesome). Two beginner's hacks have been kindly created to introduce you to Sonic Pi and Mcpi via Ben Davies.

For those of you who want to stretch beyond Minecraft then we will have a dedicated Pi hacks section with hacks that have no involvement with Minecraft. This will be updated as time goes by.

For those of you who are teachers who intend to use this or part of it for teaching programming concepts. Then there is a hack pack skills check sheet out in the next few weeks, so students can track their development over time.

Hack 1: Steve's First Musical Steps by : @b3ndavi3s

In this activity you will write a program that plays notes and sets blocks

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and select Start Game	Open Sonic Pi (menu - programming - sonic pi)
λ get the player's position λ set a block λ post messages λ create loops λ use delays to sequence the events λ use variables	Select an existing world or Create a New World Use the tab key to release the cursor from minecraft	You will need to use sonic pi 2.6 to get this type the following into the lx terminal - sudo apt-get update && sudo apt-get install sonic-pi Select a new workspace (buffer)

What are you trying to do?	Your pseudo-code

mc_message "Steve's First Musical Steps"
loop do
 g = 60
 use_synth :zawa
 5.times do
   x, y, z = mc_location
   play g
   mc_set_block :glowing_obsidian, x, y, z
   g=g+2
   sleep 0.5
 end
end
 	
#click on run to play your program
	Posts message to screen
	creates a forever loop
	set variable g as 60
	selects synth zawa
	repeat 5 times
	sets the player's current location as x, y, z
	plays note 60
	sets block at player's current position to glowing obsidian
	increases variable g by 2
	delays for 0.5 secs
	closes the repeat 5 times loop
	closes the forever loop

Challenges
- change the synth
- change the block
- change the numbers of times the loop repeats - what do you notice? can you explain why this happens?

Questions to answer about what you have coded and reflect on what you have learned:

 Code concepts used?

 What role do they perform in the script?

 Errors encountered and solutions?

 How could you extend this script?

Hack 2:The Man Who Fell To Earth (Ben Davies) @b3ndavi3s

In this activity you will write a program that creates a melody and teleports Steve

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and select Start Game	Open Sonic Pi (menu - programming - sonic pi)
λ get the player's position λ post messages λ create loops λ use delays to sequence the events λ use variables λ uses threads λ play a melody	Select an existing world or Create a New World Use the tab key to release the cursor from minecraft	You will need to use sonic pi 2.6 to get this type the following into the lx terminal - sudo apt-get update && sudo apt-get install sonic-pi Select a new workspace (buffer)

What are you trying to do?	Your pseudo-code

Sonic Pi Code

Code	Comment
mc_message "Time to teleport"	Posts message to screen
sleep 2	delays the program by 2 secs
in_thread do	creates a thread
3.times do	
use_synth :mod_dsaw	selects the synth to use
use_bpm 90	sets the bpm
3.times do	sets the notes to play and the length
play 69, release: 0.5	
sleep 0.5	
end	
play 67, release: 0.5	
sleep 1	
play 69, release: 0.33	
sleep 0.33	
play 70, release: 0.33	
sleep 0.33	
play 69, release: 0.33	
sleep 0.33	
play 67, release: 0.5	
sleep 0.5	
end	
end	
mc_teleport 27, 550, 12	teleports the player to the given coordinates
x = 10	sets the variable x to 10
10.times do	repeat 10 times
mc_message x	posts the value of variable x to screen
sleep 1	delays the program by 1 secs
x=x-1	decreases variable x by 1
end	closes the repetition indent
mc_message "Welcome back to the Ground	posts a message to screen

Challenges

- λ   change the melody
- λ   change the position to teleport to
- λ   change the countdown

λ   change the messages posted to screen

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend </u>this script?

Hack 3:Towers of Randomness (Ben Davies)

In this activity you will write a program that creates towers of random blocks

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and select Start Game	Open Sonic Pi (menu - programming - sonic pi)
• get the player's position	Select an existing world or Create a New World	You will need to use sonic pi 2.6 to get this type the following into the lx terminal - sudo apt- get update && sudo apt-get install sonic-pi
• set a block • post messages • create loops • use delays to sequence the events • use variables • use selection	Use the tab key to release the cursor from minecraft	Select a new workspace (buffer)

What are you trying to do?	Your pseudo-code

Sonic Pi Code

| ```
mc_message "Explore"
loop do
 sleep 5
 x, y, z = mc_location
 if one_in(3)
  sample :ambi_dark_woosh
  5.times do
   mc_set_block :diamond_block , x, y, z+3
   mc_set_block :glowing_obsidian , x, y+1, z+3
   mc_set_block :glowstone_block , x, y+2, z+3
   mc_set_block :gold_block , x, y+3, z+3
   y=y+4
  end
 else
  sample :ambi_haunted_hum
  5.times do
   mc_set_block :tnt , x, y, z+3
   mc_set_block :lapis_lazuli_block , x, y+1, z+3
   mc_set_block :brick , x, y+2, z+3
   mc_set_block :redstone_ore , x, y+3, z+3
   y=y+4
  end
 end
 sleep 2
end
``` | Posts message to screen<br>creates a forever loop<br>delays the program by 5 secs<br>gets the player's current position<br>creates the probability of the following event<br>selects a sample to play<br>repeat 5 times<br>places the named-block in the given positions<br><br><br><br><br><br>increases the value y by 4<br>closes the repetition loop<br>If the probability of the previous events isn't met<br>selects a sample to play<br>repeat 5 times<br>places the named-block in the given position<br><br><br><br>increases the value y by 4<br>closes the repetition loop<br>closes the selection indentation<br>delays the program for 2 seconds<br>closes the forever loop |

Challenges

- change the samples used
- change the height of the towers
- change the types of blocks in the towers

- change the probability of the if event occurring

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

Hack 4:'ForceField' written by : (@piinthesky2015)

This hack is makes the most of the a while loop to help Steve put up a force field of air if he detects obstructions in front.

This is a screen shot of the code in action

Here is the code used to create it:

```
import mcpi.minecraft as minecraft
import time
from mcpi import block
mc=minecraft.Minecraft.create()

def forcefield():
    pos = mc.player.getPos()
    mc.setBlocks(pos.x-3, pos.y, pos.z-3, pos.x+3, pos.y+3, pos.z+3, block.AIR)

while True:

    pos = mc.player.getTilePos()
    air = mc.getBlock(pos.x+1, pos.y+1, pos.z+1)
    if air != block.AIR.id:
        mc.postToChat ("Forcefield Active")
        forcefield()
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

Hack 5: CrazyTown library how to use (CrazySqueek)

Picture of library call scripts in action

Crazy Squeek is a 9 year old programming wonder who lives in the UK. He has developed a set of functions that create a series of buildings ranging from cottages to tower blocks. The purpose of this tutorial is not to write the code out but download and better understand the following:

- What libraries are ?
- How the are called ?
- Why they are useful ?
- How to use functions from libraries in your program
- You will also learn how to make effective comments. This kid is good :)

1. Step download the 3 python files needed from the below links
2.Open 'basicdemo.py'
3. Open minecraft create a new world.
4. Press f5 to run the program
5. Watch :)
6. Open Town.py choose one additional function that you can try and call in the basic demo file.

7. Try and call this in the 'basic demo.py file see if it compiles and then watch it build your additional building.

8. Now open 'flatsdemo.py' create a new world and repeat the process from tasks 6/7.

Challenge yourself

9. See if you can add in a loop to create multiple copies of the same building as used by crazysqueek.

You can now download all three python files from here https://goo.gl/Kj7pJ2 on the lesson 8 section.

Questions to answer about what you have coded and reflect on what you have learned:
 Code concepts used?

 What role do they perform in the script?

 Errors encountered and solutions?

How could you extend this script?

Hack 6:Building a skywars level with Code written by : @ncscomputing

Pic1 of platforms

Instructions

1. Type the following code in python IDE

2. Open a new Minecraft world

3. Press f5 and compile the code to check for logic and sytax errors.

4. Ensure that steps() function call toward the bottom of the code is commented out you will need this in a bit.

5. To create the skywars type sky platforms you will need to walk run the code and walk around your map. Your program should create a platform every 8 seconds. The default height is set to ten bocks high. You should end up with something that resembles pic1.

Push yourself

Can you extend the code to randomly place platforms a different heights from 10-35 blocks?

6. In the main while loop comment out the platform() and uncomment the steps() to enable it to work.

7. This will create glass steps to help you get across from platform when playing the game. You should run the code and walk around linking the platforms together like pic2.

Pic 2 of platforms with added walk ways using second function

Code:

```
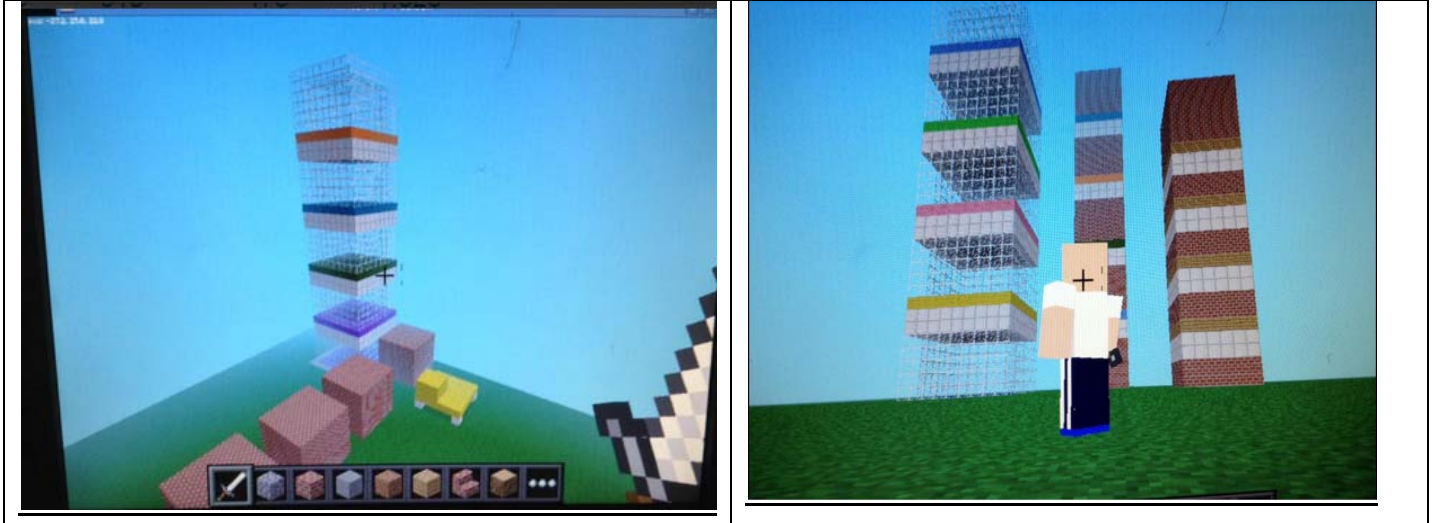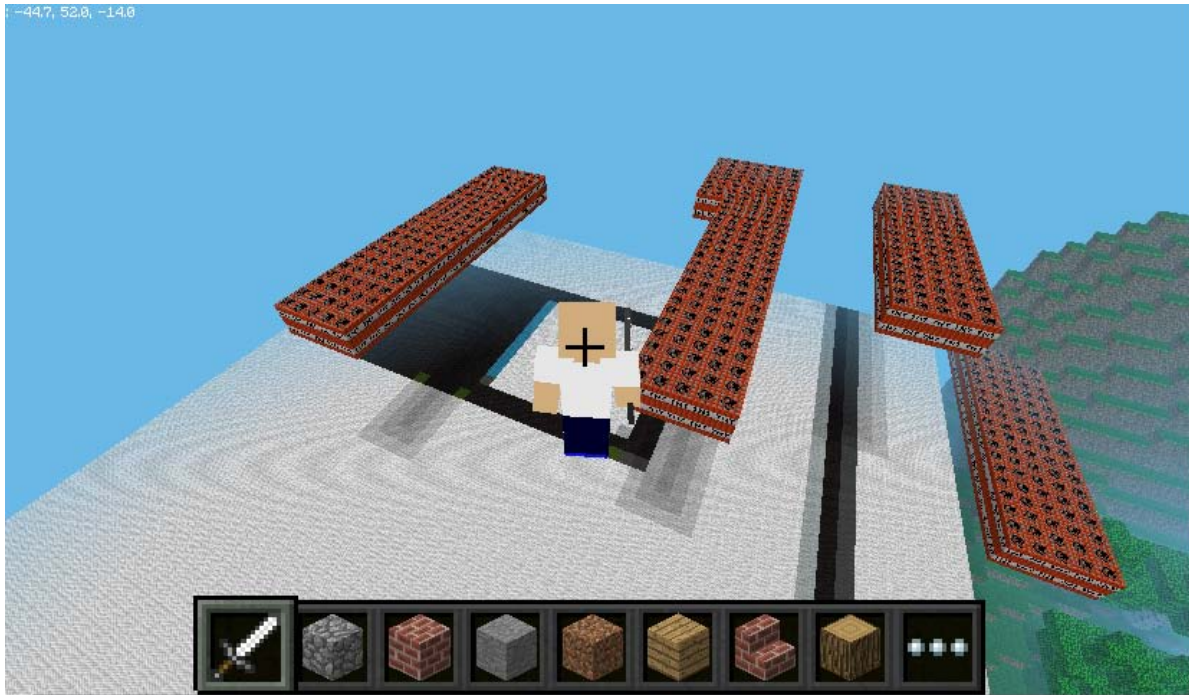import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time

mc = minecraft.Minecraft.create()
Height = 10
def platform(Height):
    blockId = 46
```

```
    x,y,z = mc.player.getPos()
    y=y+Height
    mc.setBlocks(x,y,z,x+4,y+1,z+20,blockId)
    time.sleep(8)



def steps():
    x,y,z = mc.player.getPos()
    mc.setBlock(x,y-1,z,20)
    time.sleep(0.2)



while True:
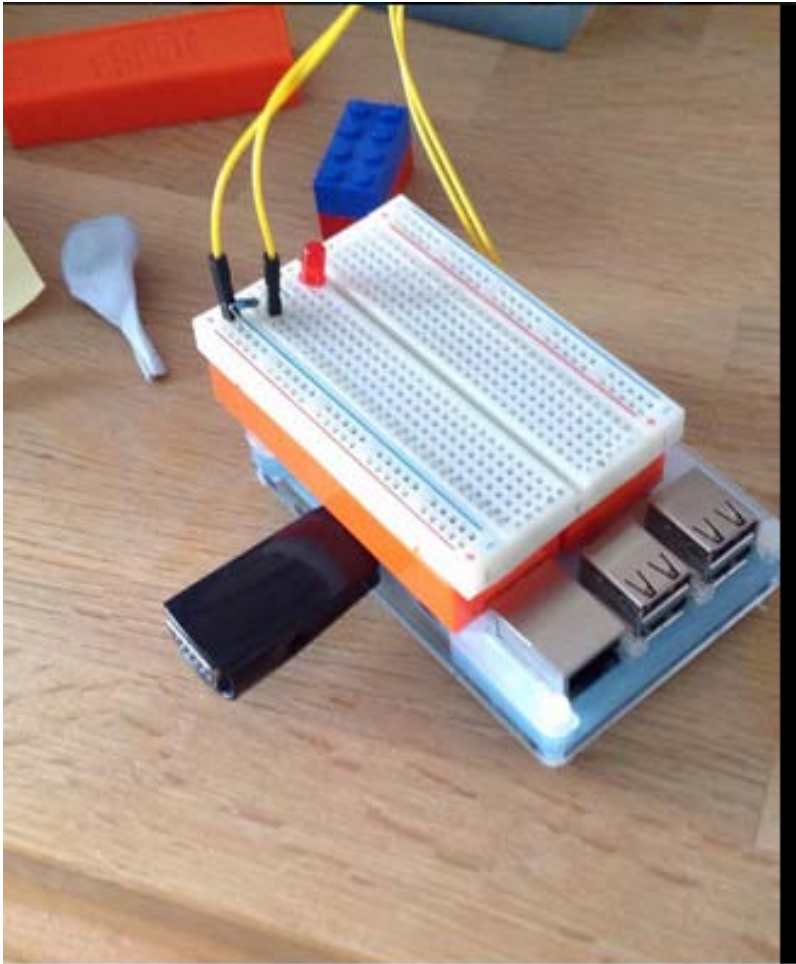    platform(Height)
    #steps()
```

Hack 7 Whack a selfie with Raspi2png and Camjam Edukit 1 @ncscomputing

This hack has used the Cam Jam Edu kit one. Which is a available to buy from here:
http://thepihut.com/products/camjam-edukit

It was my first real attempt at using breadboards. The basic concept of this hack is to do the following:

- Install 'raspi2png', which is a bit of software that has been created to allow you to take screenshots of minecraft pi

- Using the Cam Jam Edu Kit 1 breadboard to install an LED that lights up when you have taken a minecraft screen shot by whacking the TNT block.

- So when you place a TNT block you can use the whack a block code from volume one to allow Steve to whack the block and take a screen shot.

- This achieved by running something called 'shell script'. This is essentially running Linux LX terminal commands from within the python script.

- The basic set up is pictured below:

You will need to download the software by working through the blog post by Les Pounder, which can be found [here](#).

The set up for the breadboard is a mirror of the 'LED' work sheet two described in the brilliant Cam Jam resource pack here is the link https://goo.gl/Dlrd0y. N.b. I have only used one LED, they use three ☺.

Once you have your breadboard set up and raspi2png set up(I had the software folder on my pi desktop) you will need to write the code below.

Finally you will need to save the py file in the same folder as raspi2png software and use LX terminal to run the script. This is to do with using the GPIO pins. This is covered very clearly in the Cam Jam resource. ☺ Now go snap yourself and what you create.

Code

import subprocess

```python
import RPi.GPIO as GPIO

import time

import mcpi.minecraft as minecraft

import mcpi.block as block


mc = minecraft.Minecraft.create()


mc.postToChat("Whack a tnt block to take a selfie")

def On():

    GPIO.setmode(GPIO.BCM)

    GPIO.setwarnings(False)

    GPIO.setup(18,GPIO.OUT)

    print "Lights on"

    GPIO.output(18,GPIO.HIGH)


def Off():

    GPIO.setmode(GPIO.BCM)

    GPIO.setwarnings(False)

    GPIO.setup(18,GPIO.OUT)

    print "Lights off"

    GPIO.output(18,GPIO.LOW)

    GPIO.cleanup()
```

```
while True:

  evs = mc.events.pollBlockHits()


  for e in evs:

    pos = e.pos


    b = mc.getBlock(pos.x,pos.y,pos.z)


    if b == 46:

      mc.postToChat("Smile")

      On()

      time.sleep(3)

      a = subprocess.check_output('./raspi2png -d 3 -p "1.png"',shell=True)

      Off()
```

Questions to answer about what you have coded and reflect on what you have learned:

 Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

Hack 8: Steves Horizontal escalator @ncscomputing

Pic1 of code working

<u>**Instructions**</u>

Build a small square track of TNT and then write the following code. You should get Steve to move around the basic track.

Code so far:

```
import mcpi.minecraft as minecraft
import mcpi.block as block
import time

LineColour = 46
mc = minecraft.Minecraft.create()

def Main():
    x,y,z = mc.player.getPos()

    #get block -1
    CurrentBlock = mc.getBlock(x,y-1,z)
    #block2 == mc.getBlock(x+1,y-1,z)
    #Go straight ahead
```

```
     #if block -1 == 46 then
     if CurrentBlock == 46 & mc.getBlock(x,y-1,z-1)== 46:#1ststraight
        mc.player.setPos(x,y,z-1)



     elif CurrentBlock == 46 & mc.getBlock(x+1,y-1,z)== 46:#1strightcrnr
        mc.player.setPos(x+1,y,z)
           #ask is next block in frnt 46



     elif CurrentBlock == 2 & mc.getBlock(x,y-1,z+1)== 2:#2ndrightcrnr
        mc.player.setPos(x,y,z+1)


     elif CurrentBlock == 2 & mc.getBlock(x-1,y-1,z)== 2:#3ndrightcrnr
        mc.player.setPos(x-1,y,z)
while True:
     Main()
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend </u>this script?

Code so far:

```
import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time

mc = minecraft.Minecraft.create()
pos = mc.player.getPos()
def SetRoad():
    pos = mc.player.getPos()
    mc.setBlock(pos.x, pos.y-1, pos.z, 35,1)#orange
    mc.setBlock(pos.x-1, pos.y-1, pos.z, 35,2)#pinky purple
    mc.setBlock(pos.x-2, pos.y-1, pos.z, 35,3)#sky blue
    mc.setBlock(pos.x-3, pos.y-1, pos.z, 35,4)#yellow
    mc.setBlock(pos.x-4, pos.y-1, pos.z, 35,5)#green
```

```
    mc.setBlock(pos.x-5, pos.y-1, pos.z, 35,6)#pink
    mc.setBlock(pos.x-6, pos.y-1, pos.z, 35,7)#black
    mc.setBlock(pos.x-7, pos.y-1, pos.z, 35,8)#grey
    mc.setBlock(pos.x-8, pos.y-1, pos.z, 35,9)#blue
    mc.setBlock(pos.x-9, pos.y-1, pos.z, 35,10)#purple


#mc.postToChat("Watch out for disco blocks")



while True:

    SetRoad()
    time.sleep(0.05)
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you extend this script?

Hack 10: Text Messages V1 written by : @ncscomputing

This takes the concept of making a wall and displaying letters inside this wall. The key concepts covered are using multi dimensional lists and how to update them. It also covers functions and returning values with a function.

Pic1 of code working

Code

```
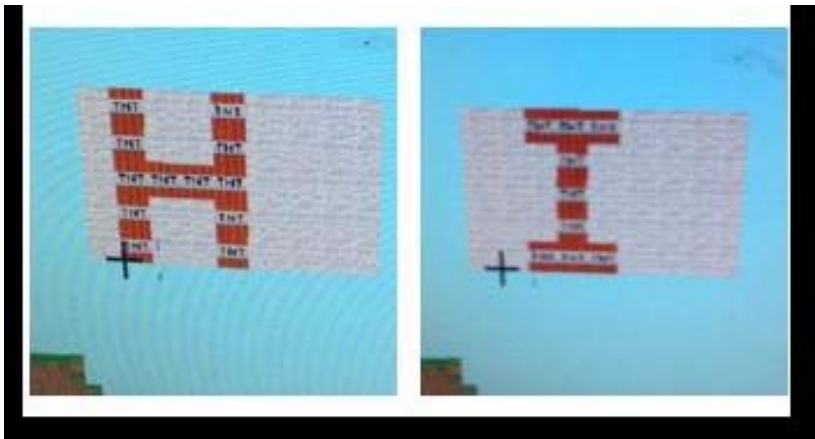import mcpi.minecraft as minecraft

import mcpi.block as block

import random

import time



mc = minecraft.Minecraft.create()



def LetterI():
```

```python
    row1 =[35,35,46,46,46,35,35]

    row2 = [35,35,35,46,35,35,35]

    row3 = [35,35,35,46,35,35,35]

    row4 = [35,35,35,46,35,35,35]

    row5 = [35,35,46,46,46,35,35]

    TempList = [row1,row2,row3,row4,row5]

    return TempList


def LetterH():


    row1 = [35,46,35,35,46,35,35]

    row2 = [35,46,35,35,46,35,35]

    row3 = [35,46,46,46,46,35,35]

    row4 = [35,46,35,35,46,35,35]

    row5 = [35,46,35,35,46,35,35]

    TempList = [row1,row2,row3,row4,row5]

    return TempList


def PrintWall(ImportedList):


    pos = mc.player.getTilePos()

    mc.player.setPos(pos.x,pos.y,pos.z)
```

```python
myList = ImportedList


for row in range (0,5):

    for column in range (0,7):


        mc.setBlock(pos.x+column,pos.y+row,pos.z-20,myList[row][column])



while True:

    PrintWall(LetterI())

    time.sleep(2)

    PrintWall(LetterH())

    time.sleep(2)
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

81

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

Hack 11: Sky wars / splatoon 60 second painting game: Game logic written by :
@ncscomputing

Status : bugs with timer and point scoring, TO BE FIXED WHEN POSSIBLE

Pic1 of platforms which have been changed orange by the player walking over them.(The aim of the game is to colour as many TNT blocks orange as possible in 1 minute)

Code so far

```
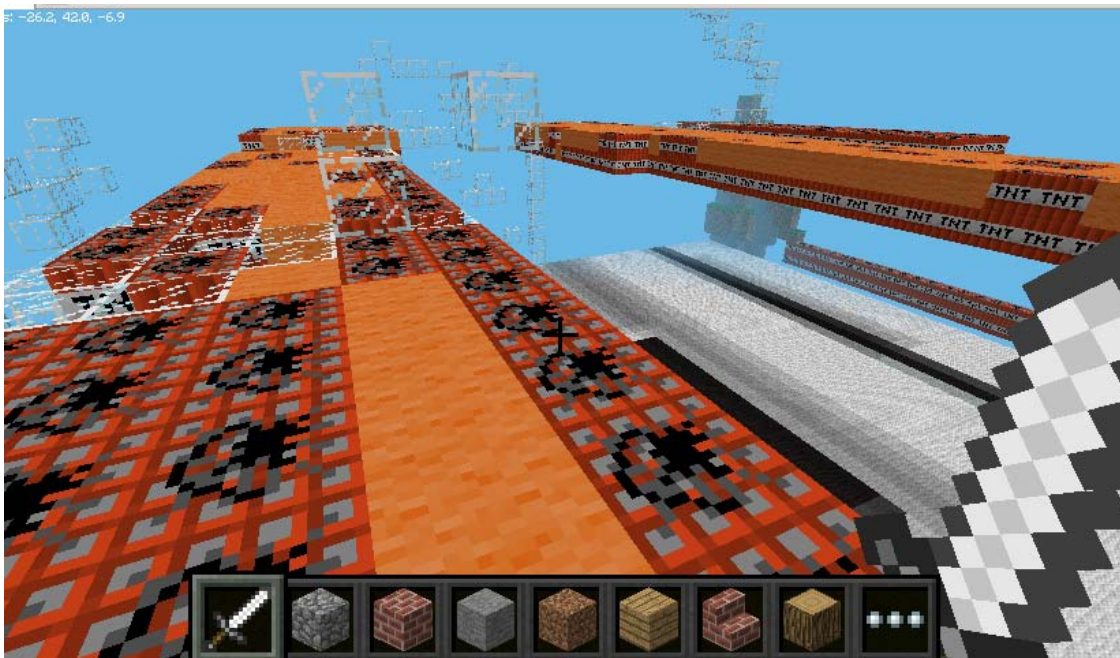import mcpi.minecraft as minecraft
import mcpi.block as block
import random
```

```python
import time


Score = 0
TeamColour = 5#Green
count = 0
mc = minecraft.Minecraft.create()
Height = 10


def Timer():
    now = time.localtime(time.time())
    return now[5]


def platform(Height):
    blockId = 46
    x,y,z = mc.player.getPos()
    y=y+Height
    mc.setBlocks(x,y,z,x+4,y+1,z+20,blockId)
    time.sleep(8)


def steps():
    x,y,z = mc.player.getPos()
    mc.setBlock(x,y-1,z,20)
    time.sleep(0.2)

def BasicGameLogic(Score):

    x,y,z = mc.player.getPos()
    block = mc.getBlock(x,y-1,z)
    if block == 46:
        mc.setBlock(x,y-1,z,35,1)#Orange Wool
        Score = Score+1
        print "score:",Score
```

```
        #mc.postToChat("Tnt")


#mc.player.setPos(pos.x,pos.y,pos.z)
#mc.camera.setPos(pos.x,pos.y,pos.z)
#mc.camera.setFollow()
CurrentSec = 0
while CurrentSec <=60:

  # platform(Height)

  #steps()
  BasicGameLogic(Score)
  CurrentSec = Timer()
  print CurrentSec
  if CurrentSec == 59:
      print "Time over, your Score is: ", Score
      break
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend </u>this script

Hack 12 : Minecraft Photo Booth (Dan Aldred)

This hack creates a photobooth in Minecraft which when you the player walk in, it triggers the Pi Camera and takes your picture, awesome!

Add your Pi Camera and boot up your Pi, load Python and also Minecraft.

```
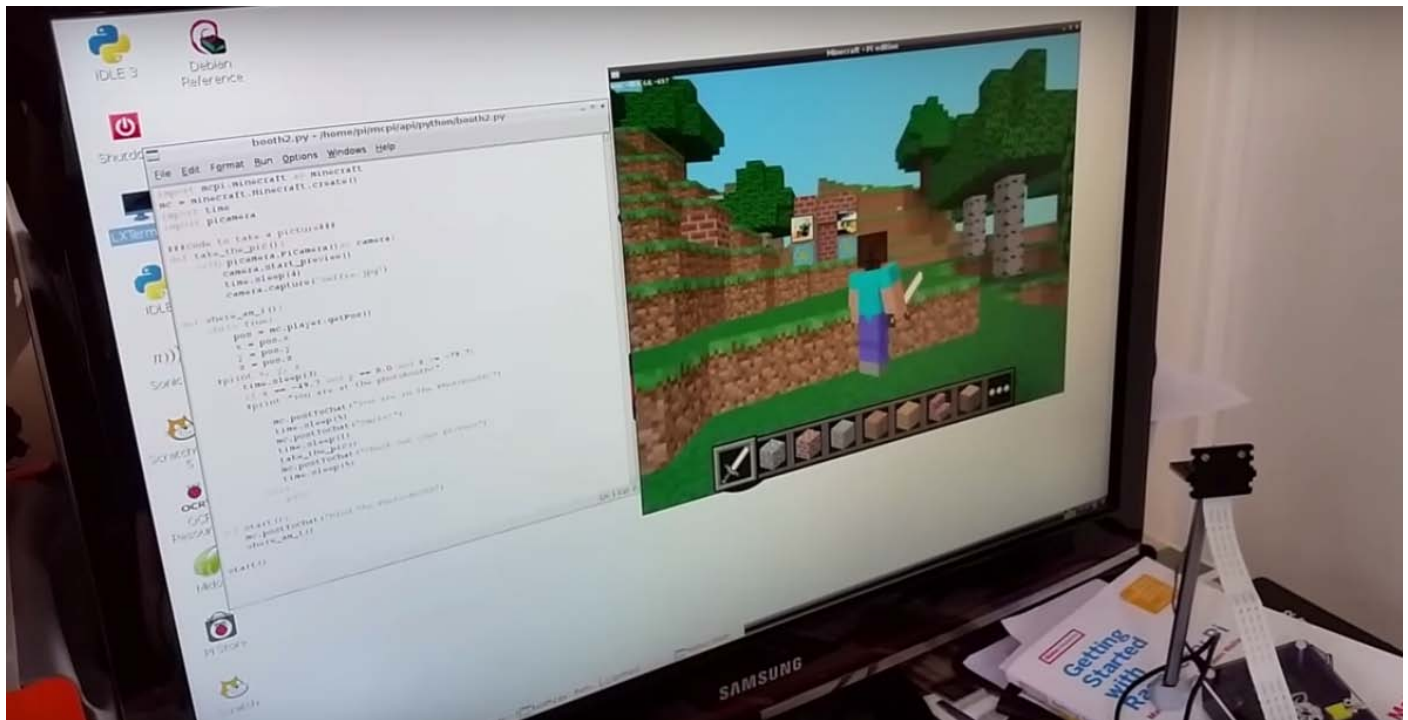import mcpi.minecraft as minecraft
mc = minecraft.Minecraft.create()
import time
import picamera

###Code to take a picture###
###@TeCoEd###
def take_the_pic():
    with picamera.PiCamera()as camera:
        #camera.resolution = (150, 100)
        camera.start_preview()
        time.sleep(2)
```

```python
        camera.capture('selfie.jpg')


def where_am_I():
    while True:
        pos = mc.player.getPos()
        x = pos.x
        y = pos.y
        z = pos.z
    #print x, y, z
        time.sleep(3)
        if x >= 10.5 and y == 9.0 and z == -44.3:
        #print "You are at the photobooth!"

            mc.postToChat("You are in the Photobooth!")
            time.sleep(1)
            mc.postToChat("Smile!")
            time.sleep(1)
            take_the_pic()
            mc.postToChat("Check out your picture")
            time.sleep(5)
        else:
            pass

def start():
    mc.postToChat("Find the Photo-Booth")
    where_am_I()

start()
```

Questions to answer about what you have coded and reflect on what you have learned:

 Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend </u>this script?

Hack 13 : Minecraft Mine-Sweeper (Dan Aldred)

You may remember or have even played the classic Minesweeper game which dates back to the 1960's as one of the earliest mainframe computer games. Over the years it has been bundled with most operating systems and even featured as a mini game variation on the New Super Mario Bros. In this hack you will ceate a simple version in Minecraft, yes a Minecraft Mine-Sweeper!


```
###TeCoEd###
###MineCraft Sweeper###

import random
import time
from mcpi import minecraft
```

```
mc = minecraft.Minecraft.create()

###Creates the Board###
mc.postToChat("Welcome to Minecraft MineSweeper")
x, y, z = mc.player.getPos()
mc.setBlocks(x, y-1, z, x+20, y-1, z+20, 58)

global mine
mine = random.randrange(0, 11, 1)
print mine

###Places the mine###
mine_x = int(x+mine)
mine_y = int(y-1)
mine_z = int(z+mine)

print mine_x, mine_y, mine_z ###test
mc.setBlock(mine_x, mine_y, mine_z,58)

score = 0

mc.postToChat("Score is "+str(score))
#test = mc.setBlock(x + mine, y-1, z + mine, 46,1)
time.sleep(10)
while True:  ###TEST IF YOU STAND ON THE BLOCK

    x1, y1, z1 = mc.player.getTilePos()
    #print x1, y1, z1 ###test
    time.sleep(0.1)
    score = score + 1
    if (x1, y1-1, z1) == (mine_x, mine_y, mine_z):
        mc.setBlocks(x-5, y+1, z-5, x+5, y+2, z+5, 10) ##CHANGE TO WATER?
        print "GAME OVER"
        mc.postToChat("G A M E  O V E R")
        mc.postToChat("Score is "+str(score))
        break
```

```
else:
    mc.setBlock(x1, y1-1, z1, 41)
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you extend this script?

Hack 14:Unicorn HAT Element Finder (TeCoEd) written by : @dan_aldred

Stand on a block, the Unicorn responds with the relevant colour

```python
 #!/usr/bin/env python

import unicornhat as UH
import time
import sys
from mcpi import minecraft
mc = minecraft.Minecraft.create()

UH.brightness(0.10)

def water():
 for y in range(8):
  for x in range(8):
    UH.set_pixel(x,y, 0, 0, 255)
```

```
      UH.show()
      time.sleep(0.02)

def air():
  for y in range(8):
    for x in range(8):
      UH.set_pixel(x,y, 0, 0, 0)
      UH.show()
      time.sleep(0.02)

def TNT():
  for y in range(8):
    for x in range(8):
      UH.set_pixel(x,y, 255, 0, 100)
      UH.show()
      time.sleep(0.02)


def dirt():
  for y in range(8):
    for x in range(8):
      UH.set_pixel(x,y, 0, 255, 0)
      UH.show()
      time.sleep(0.02)

def sand():
  for y in range(8):
    for x in range(8):
      UH.set_pixel(x,y, 148, 0, 211)
      UH.show()
      time.sleep(0.02)



while True:
  x,y,z = mc.player.getPos()

  blockID =  mc.getBlock(x, y-1, z)
  print blockID

  time.sleep(0.1)
```

94

```
if blockID == 9:
    water()
elif blockID == 0:
    air()
elif blockID == 2:
    dirt()
elif blockID ==12:
    sand()
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you _extend_ this script? Try adding your own blocks and colours

Hack 15:Sense Hat Interactive House V1 written by : @ncscomputing

This hack uses the Astro Pi hat or Sense hat to introduce how to interact Minecraft with Physical data such as Temperature.

Pics

Code

```
import mcpi.minecraft as minecraft
import mcpi.block as block
from astro_pi import AstroPi
import time


mc = minecraft.Minecraft.create()
ap = AstroPi()
TheTime = time.ctime()

while True:

    evs = mc.events.pollBlockHits()
    for e in evs:
        pos = e.pos
        block = mc.getBlock(pos.x,pos.y,pos.z)

        if block == 46:
            temp = int(ap.get_temperature())
            msg = "The temp is:", temp
            mc.postToChat(msg)
```

Questions to answer about what you have coded and reflect on what you have learned:

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend </u>this script?

Hack 16: A flat map download for Raspberry Pi Minecraft by Wizard Keen

This is a flat map that you can download and load into your worlds on the Raspberry Pi. Click / type in the shortened link below to download from his dropbox.

https://goo.gl/KzKI0N

Volume 3
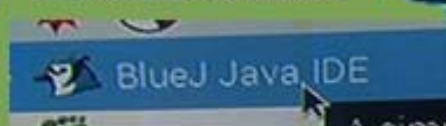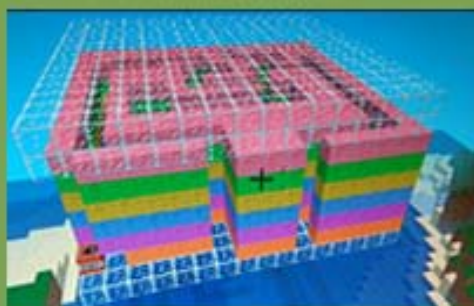
Hack 1: Rainbow Road 2.0, bigger road, more efficient code by @ncscomputing

It's back, but wider and with less code overall. It solves the problem with more efficient code.

Code in action
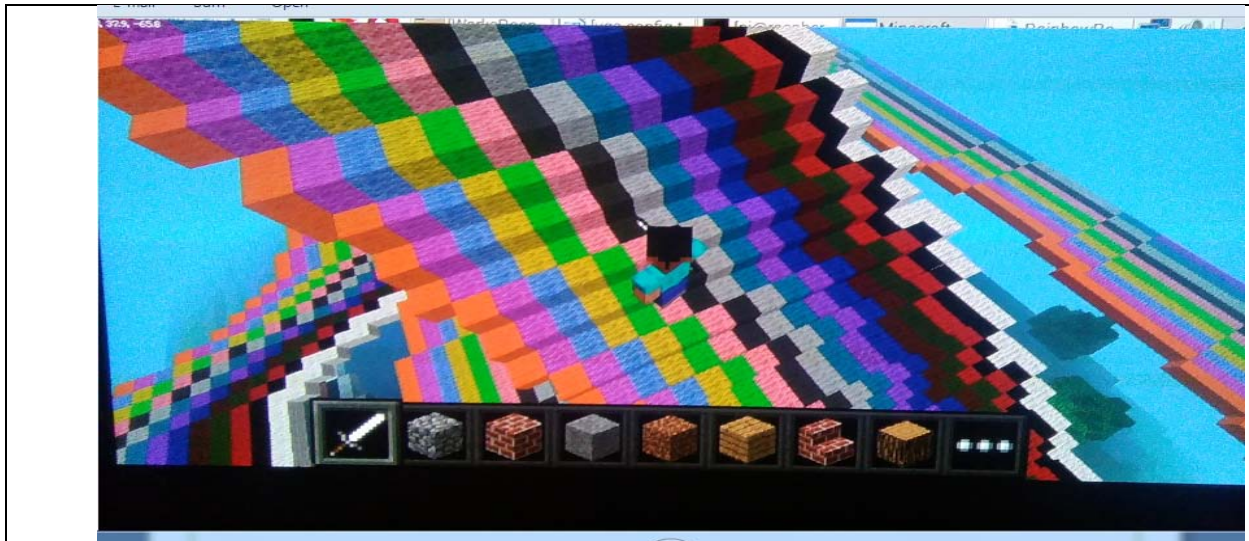
Code

```
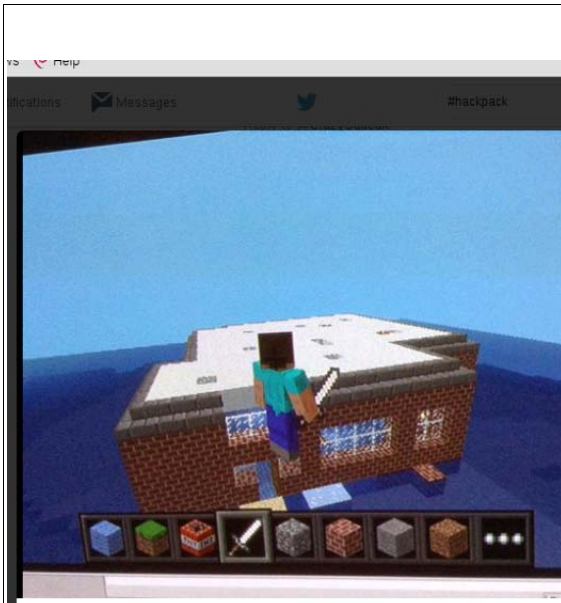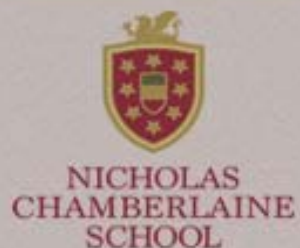import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time


mc = minecraft.Minecraft.create()

pos = mc.player.getPos()

def SetRoad() :
    count = 1
    pos = mc.player.getPos()
    while count <= 16:
        mc.setBlock(pos.x-count,pos.y-1,pos.z,35,count)
        count = count +1

mc.postToChat("Rainbow Road 2.0!")

while True:
    SetRoad()
#    time.sleep(0.1)
```

102

<u>Questions to answer about what you have coded and reflect on what you have learned:</u>

Code concepts used?

What role do they perform in the script?

Errors encountered and solutions?

How could you <u>extend</u> this script?

Hack 2: Rainbow Road: randoms by @ncscomputing

It's back, but wider and with less code overall, now random blocks too using a list to generate random block Id's.

Code in action

Code

```
File  Edit  Format  Run  Options  Windows  Help
import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time


mc = minecraft.Minecraft.create()

pos = mc.player.getPos()

BlockIdsList=[1,2,7,21,22,31,46,49,73,80,246,247,321,112]

def SetRoadWool():
    count = 1
    pos = mc.player.getPos()
    while count <= 16:
        mc.setBlock(pos.x-count,pos.y-1,pos.z,35,count)
        count = count +1

def SetRoadRandom():
    count = 1
    pos = mc.player.getPos()
    while count <= 16:
        mc.setBlock(pos.x-count,pos.y-1,pos.z,random.choice(BlockIdsList))
        count = count +1



mc.postToChat("Rainbow Road 2.0!")

while True:
    SetRoadWool()
    SetRoadRandom()
#   time.sleep(0.1)
```

Questions to answer about what you have coded and reflect on what you have learned:

 Code concepts used?

What role do they perform in the script?

 Errors encountered and solutions? How could you <u>extend </u>this script?

105

Hack 3 My first Code bug program: face changer by @ncscomputing

This is a very basic intro into code bug. All I have done is created 3 sprites using blockly on the code bug website http://www.codebug.org.uk/create/codebug/3653/smiley/ and the face output on to the led screen changes according to which button is pressed.

Picture of emulation output

Steves normal face	"A" button pressed	"B" button pressed
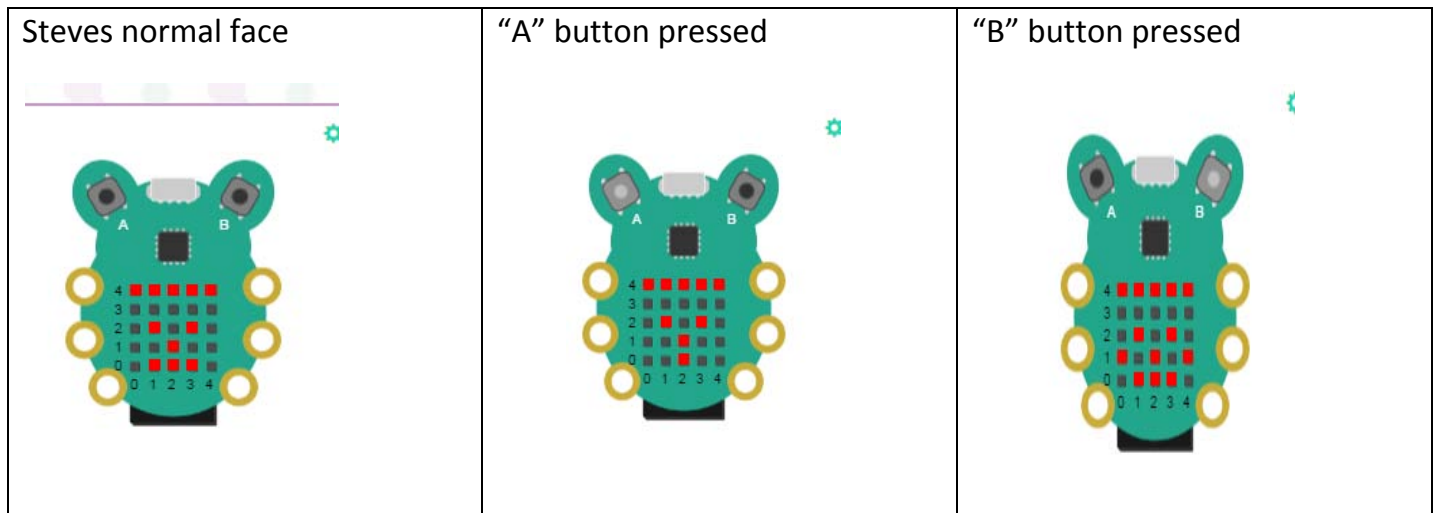
Code
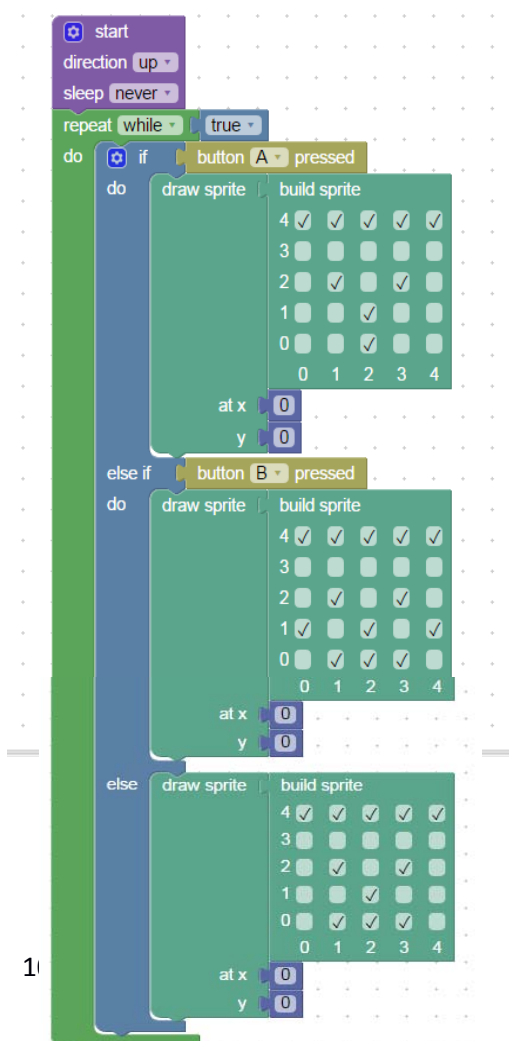
Introduction RiscOS: Simple Programming with Raspberry Pi running RISC OS by @tomW_Ident

Introduction – What is RISC OS?

RISC OS is an alternative operating system supported by the Raspberry Pi foundation for use with all versions of the Raspberry Pi computer.
(Not officially supported on the Pi Zero at time of writing)

Unlike 'Raspbian' which is the standard OS for the Pi. RISC OS is a completely standalone ARM platform, and not a form or flavour of Linux.

Instead of Python, RISC OS uses a language called BASIC as its main built in programming environment. BASIC was very popular in the 1980's with almost all home and office personal computers. Acorn Computers created one of the best versions of BASIC for their BBC and RISC OS computers known as BBC BASIC. It is a modern updated version of this language that we will be using with the Raspberry Pi.

These simple tutorials, take from 10-20 minutes each to complete and give maximum visual onscreen results for minimum coding time. Ideal for short classroom sessions!

For these demonstrations we'll assume your are using any model of the Raspberry Pi computer (except Pi Zero) with the latest version of RISC OS 5
(currently at time of writing this is: Build RC14 available free from: www.riscosopen.org)

A introduction video tutorial on using RISC OS on the Raspberry Pi can be watched online at:https://www.youtube.com/watch?v=qNhkHyWRwg

Starting a BASIC session

To enter code into RISC OS, we first need to start the BASIC Interpreter.

To do this, insert your SD card with RISC OS loaded and power up your Pi

RISC OS will take a few moments to load to a desktop that will look something like this:
(Don't worry if your desktop looks different, such as icons or colours)
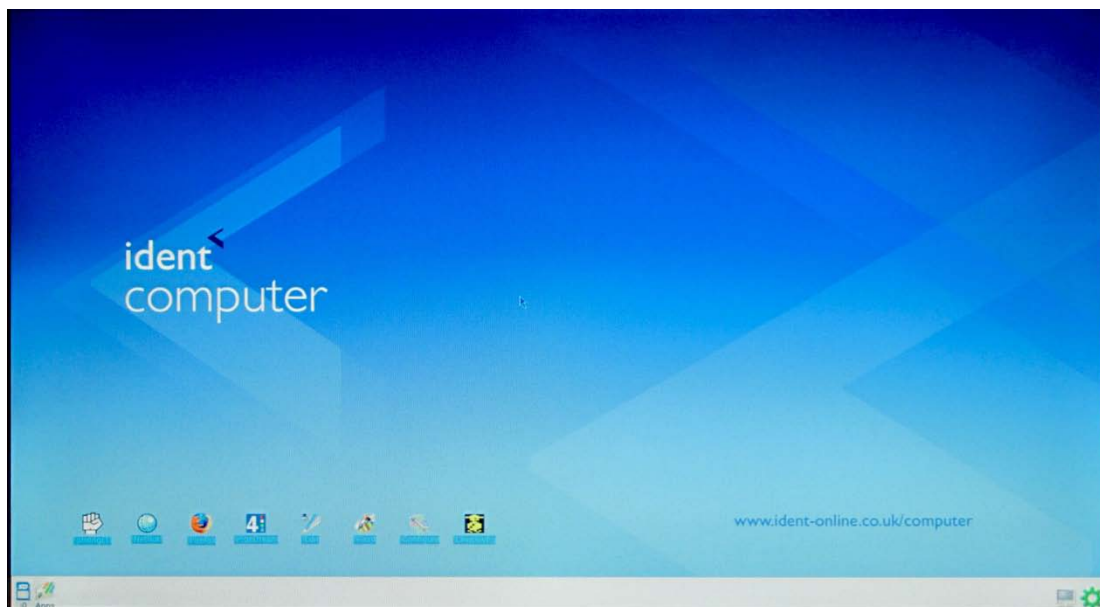
1) Once at the desktop press 'F12' once on the keyboard. This will load the 'Task Manager' which will appear as a white text box at the bottom of the screen.

2) Now type: **basic**
   Followed by pressing the enter or return key.
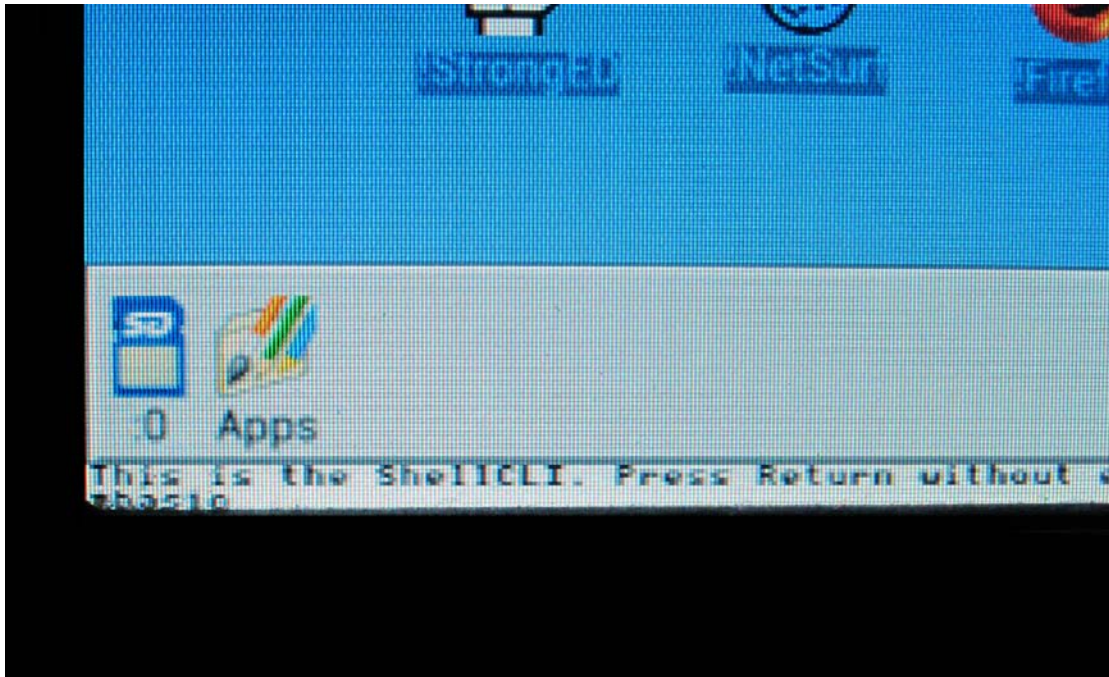
3) This will load a ARM BBC BASIC session. You are now ready to input code directly into the computer.
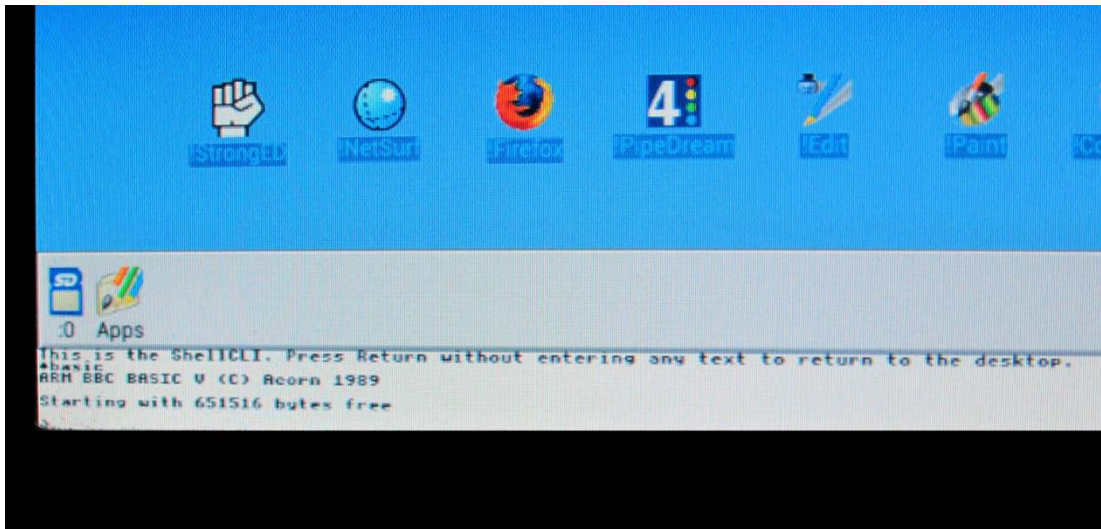
4) It is common for most programming to be written in a screen display mode, these allow different text, colours and graphics to be displayed, it also makes code easier to look at. Type: MODE1 followed by pressing the Enter or Return key.

5) After Pressing 'F12' you might have noticed your onscreen cursor disappeared and you were locked out of the desktop. If you want to end your BASIC session and return to the desktop, you can type:

 *QUIT

Followed by pressing the Enter or Return key twice.

Hack 4 with RiscOS and Basic "Hello World" by @tomW_Ident

Program 1:

Hello World!

Duration: 15 minutes approx

BBC BASIC works using a numbered list system, each line of a program is numbered. The computer will always read the lowest number first then read down with the largest numbered line forming the end of the program.
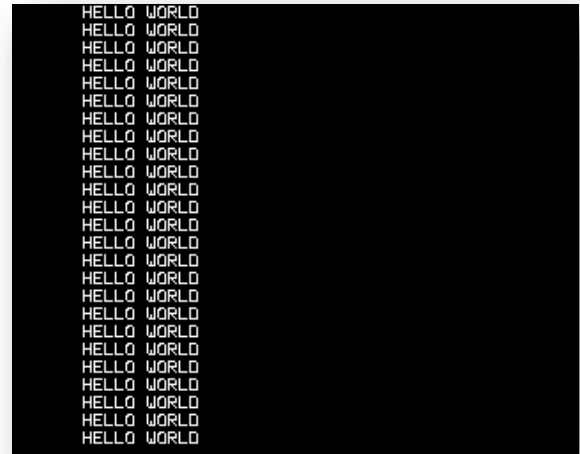
It does not matter which order you type in the program as the computer will always sort them into order with the smallest first. It is always good practice to go up in 5s or 10s when writing a program in BBC BASIC.

In this first program we'll get the computer to put the words "HELLO WORLD" on the screen, and repeat this every time we run the program.

With the **Capslock ON** type the following into the computer. Press the **'RETURN'** or **'ENTER'** key at the end of each line to type the next instruction on a new line below.

10 PRINT "HELLO WORLD!"

20 END

After line 20 press the **'RETURN'** or **'ENTER'** key again to confirm line 20. In BBC BASIC a line of code won't be entered into the computers memory until **'RETURN'** or **'ENTER'** has been pressed.

This two-line program is now in the computers memory, we can now test the program by simply typing**: "RUN"** followed by the **'RETURN'** or **'ENTER'** key.

111

The computer will run the program and should display the message "HELLO WORLD!" on the next line under the program.

If you get any other result such as **"Mistake"** or **"Syntax Error",** check you have written all the lines correctly and if needs been simply re-enter the effected line back into the computer. You can re-enter or enter a new a line of code into a program when ever you see the **">_"** prompt on the screen.

Giving a line the same number as a line already in the program will replace that exciting line of code.

Lets now clear the screen by typing: **"CLS"** followed by the **'RETURN'** or **'ENTER'** key.

The screen will now clear just leaving the **">_"** prompt. However this does not mean the computer has lost or forgotten our program.

We can check if a program is still in the computers memory by telling it to list any lines of code currently in memory. To do this we simply type: **"LIST"** followed by the **'RETURN'** or **'ENTER'** key.

The computer should now display the two lines of code once more, and if you wish, you can run the program again by typing, **"RUN"** followed by the **'RETURN'** or **'ENTER'** key.

Adding a Loop.

So far we have a simple program that prints the line "HELLO WORLD" onto the screen. We can now add a loop and tells the computer to repeat this and print the command over and over again.

To do this type the following:

15 GOTO 10

112

This will add a new line of code to our program in-between lines 10 and 20, and tells the computer to go back to line 10 and repeat that lines instruction, which is to **"PRINT"** "HELLO WORLD" onto the screen, once done, the computer will move onto line 15 again were it will be told to go back to line 10 and so on.

This will create a loop that will never end, as the computer will never get to line 20 which tells the computer to end the program, and so it will keep repeating the same two instructions.

Lets list the program again. Type **"LIST"** followed by the **'RETURN'** or **'ENTER'** key.

The program should now read as follows:

10 PRINT "HELLO WORLD"

15 GOTO 10

20 END

Now run the program by typing:

RUN

Followed by the **'RETURN'** or **'ENTER'** key.

The computer will now display a long scrolling line of words reading "HELLO WORLD" this will continue until we ask it to stop.

Pressing the **'ESCAPE'** or **'Esc'** key can stop almost any looping program written in BBC BASIC.

The computer will display a message such as **"Escaped on line 15"** followed by **">_"** prompt.

Customising the Program.

Now we have a simple looping program, we can make our own changes to it.

For example, we can retype line 10 so it reads:

10 PRINT "MY NAME IS DAVID"

Then running the program again will make the change, you could replace "David" with your own name if you want to.

With a **PRINT** command, anything we type between the two speech or quote marks will be reproduced or 'Printed' on the screen.

We could also add extra lines of code between lines 10 and 15, for example:

10 PRINT "HELLO"

11 PRINT "MY"

12 PRINT "NAME"

13 PRINT "IS"

14 PRINT "DAVID"

We don't have to retype lines 15 or 20 as they are already in the computers memory, so we only have to retype lines we want to change or type new lines in.

Remember if you type a line of code in with the same number as an existing line of code, the computer will replace that line with the new one.

With any new changes made, we can test the program again by typing **"RUN"**, clear the screen at any time by typing **"CLS"** and show the full program on screen by typing **"LIST"**. Pressing the **'RETURN'** or **'ENTER'** key follows all three commands.

Hack 5 Digital Line Art in RiscOS by @tomW_Ident
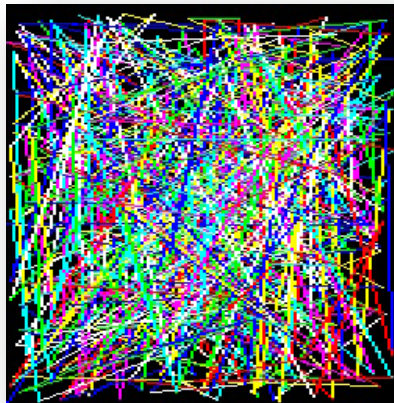
Program 2:

Digital Line Art

Duration: 20 minutes approx

In this program we take a lot of the commands we've learned and complies them into a final and very colourful display program.

Using the **"RND"** (short for 'Random') and **"DRAW"** commands combined with **"GCOL"** (Graphics Colour) and **"GOTO"** commands we'll get the computer to draw random coloured lines on the screen at totally random places and in random colours, creating a very colourful and almost artist display.

We'll also introduce **"FOR"**, **"TO"** and **"NEXT"** statements, these work together to create mini loops within our main program, key if we want tell the computer to keep drawing lines on the screen.

115

As this is a new program you'll need to start a new programming session and clear the computers memory to do this type:

Type:

NEW

Followed by the 'Enter' or 'Return' key.

As this is a graphics program we'll need to put our Pi into a graphics mode, for this example we'll use Mode 28, which is a high-resolution screen mode outputting 800x600 onscreen pixels.

Type:

MODE28

Followed by the 'Enter' or 'Return' key.

Now type in the following:

10 MODE 28

15 FOR I=1 TO 1000

20 GCOL 0,RND(64)

25 DRAW RND(1000),RND(1000)

35 NEXT I

40 PRINT "DONE"

Followed by the 'RETURN' or 'ENTER' key.

Using the "FOR" command we have also assigned a new variable called "I" which we tell the computer is equal to 1. Next in the same line we use a "TO" command followed by an end figure, in this case 1000. What this does is tell the computer to start counting and to carry out the following lines of code as many times as the latter number dictates, so in our example the computer will draw 1000 lines on the screen.

After a "FOR" and "TO" command as been called the computer starts looking for a "NEXT" statement in the following lines in our case a "NEXT" statement called "I". When it finds it, the computer then knows to loop from the "FOR" command though to the "NEXT" statement for the number of times assigned to the "TO" statement.

After the computer has looped the "FOR", "TO" and "NEXT" command for the assigned 1000 times it breaks the loop and carries out the "NEXT" command on line 35 which now simply means go to the next line, in this case line 40.

Line 40 simply prints "DONE" on the screen to tell us the program is finished and 1000 lines have been drawn on the screen.

Try out the programme by typing:

RUN

Followed by the 'RETURN' or 'ENTER' key.

Because a Raspberry Pi (even the older models) run RISC OS so fast we just get the end result and not the process of drawing the line image on screen. Adding an extra loop will slow the programme and the computer down.

Type:

30 FOR T=1 TO 10000: NEXT T

Followed by the 'RETURN' or 'ENTER' key.

Line 30 uses an additional "FOR", "TO" and "NEXT" command to created an extra looping variable called "T" (Time) which makes the computer stop and count from 1 to 10000 before continuing with the main program. This creates a pause, slowing the program down so we can see what's happening on the screen.

Try running the programme again by typing:

RUN

Followed by the 'RETURN' or 'ENTER' key.

As we've seen when the computer is finished with the looping "FOR", "TO" and "NEXT" commands it breaks and comes to the end of the program.

If we wanted to we could get the computer to keep drawing lines indefinably, until we manually stop or break the program using the 'ESCAPE' or 'Esc' key.

To create a main program loop which will make the computer start counting a whole new set of 1000 lines

To do this we can replace line 40 with:

40 GOTO 15

Followed by the 'RETURN' or 'ENTER' key.

As with our first program the "GOTO" command creates a main loop meaning the program cant ever come to the end or stop.

Type :

RUN

Followed by the 'RETURN' or 'ENTER' key.

The computer will now run this colourful display, first by running its internal program loop (up to 1000 times) then by running its main loop (starting from line 15 over again).

Remember pressing the 'ESCAPE' or 'Esc' key at any time will stop the program.

RISC OS Content, Compiled by Tom Williamson, Ident Computer

© Copyright 2015

Email: tom_williamson@rocketmail.com

Web http://www.ident-online.co.uk/computer
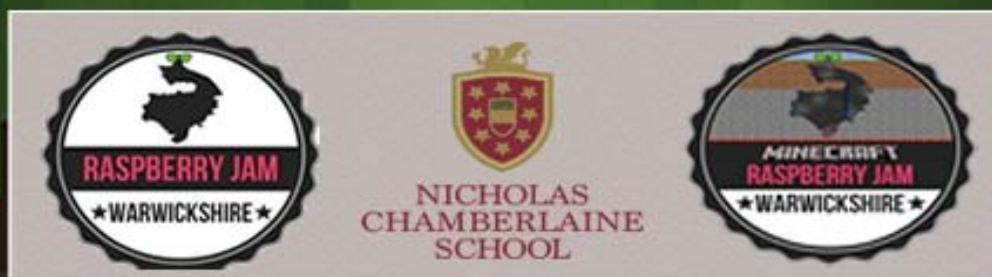
Volume 3.5

MINECRAFT COMPETITIONS MODDING CREATIVE

Hack pack Volume 3.5
Hacking Christmas

Hack 1: Blocky Selfie by @ncscomputing

Pic of hack working
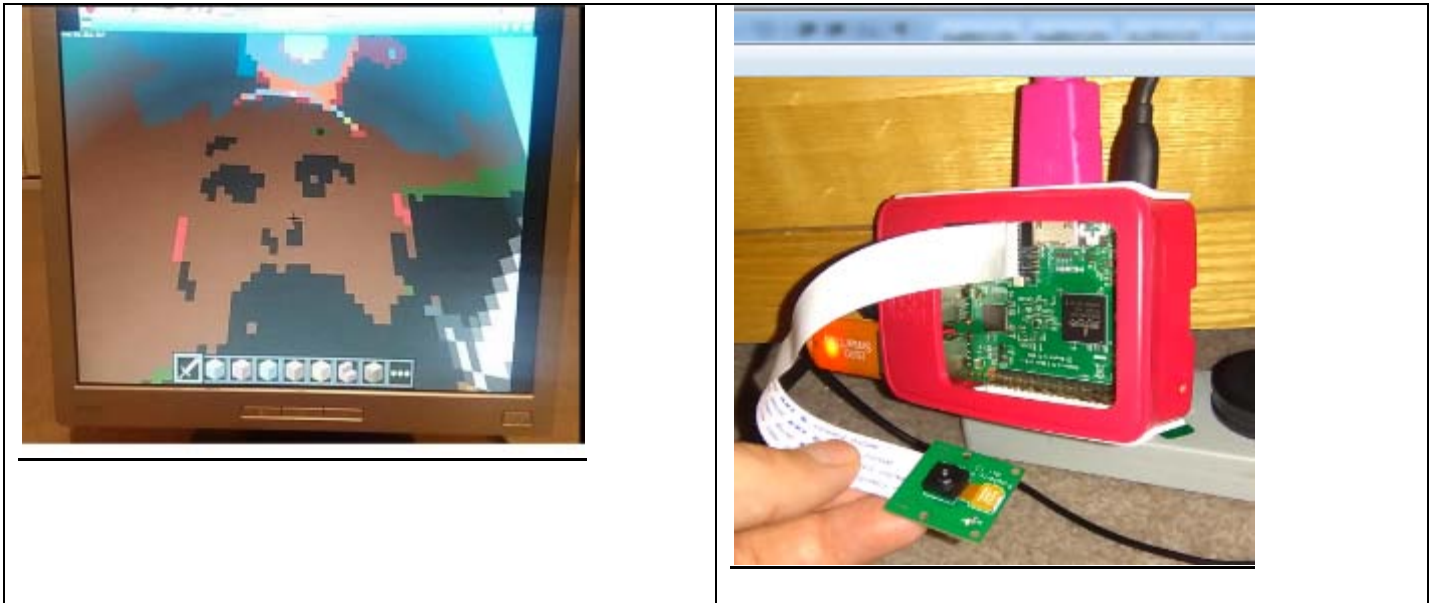
Code written by @FerriTheMaker / @martinohanlon

1. The code can be downloaded by using the following link: https://goo.gl/WcWUhq

2. Save a copy

3. Open a new minecraft world

4. Plug in the Pi Camera watch this video of how to do this https://www.raspberrypi.org/help/camera-module-setup/

5. Run the python file

6. In Minecraft right click and hit a block

7. Make sure the Pi Camera is facing you. It will take the photo (n.b. it will warn you in game)

8. Watch it render the image (TIP: try and do it in a light room/ background)

Hack 2: Pixelated Christmas Tree and Flashing lights by @ncscomputing

Pic of hack working

Code

```python
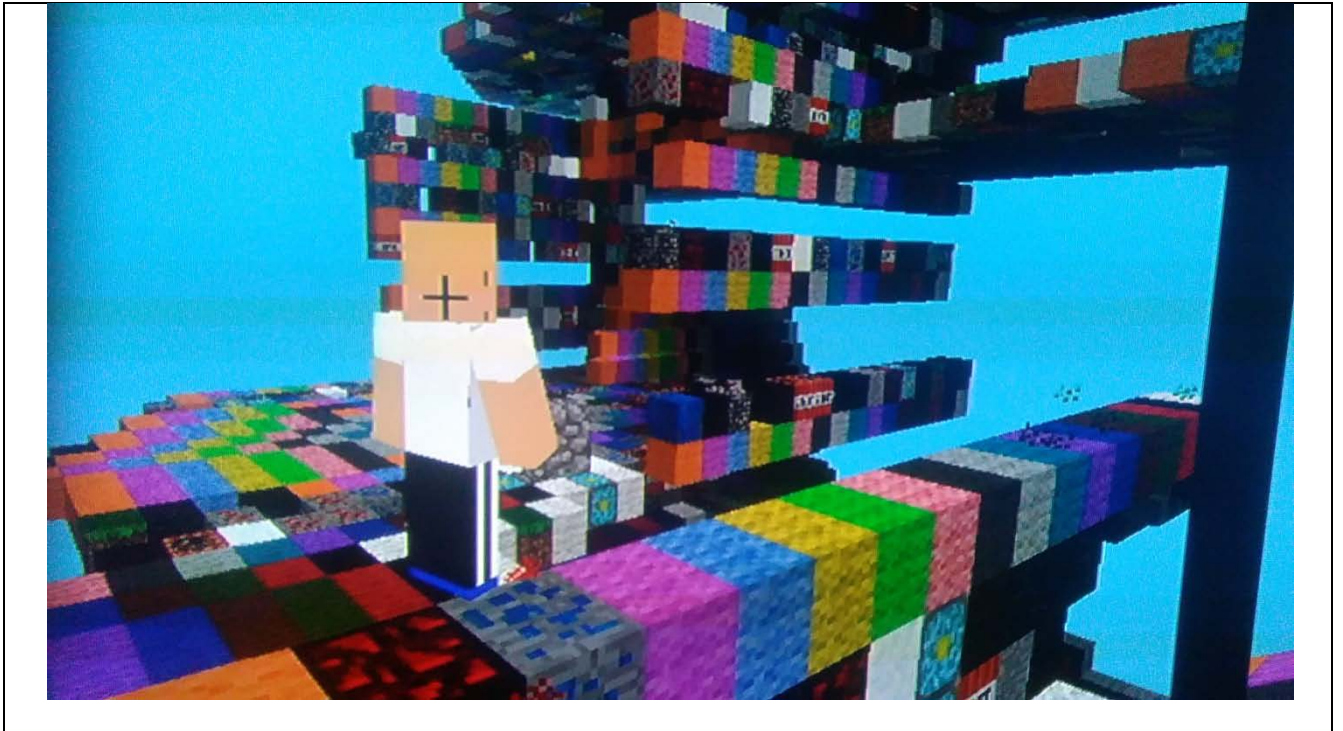import mcpi.minecraft as minecraft
import mcpi.block as block
import random
import time

BlocksList = [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16]

mc = minecraft.Minecraft.create()

def TreeBasic():

    row1 = [0,0,0,(35,random.choice(BlocksList)),0,0,0]
    row2 = [0,0,(35,random.choice(BlocksList)),18,(35,random.choice(BlocksList)),0,0]
    row3 = [0,(35,random.choice(BlocksList)),18,18,18,(35,random.choice(BlocksList)),0]
    row4 = [(35,random.choice(BlocksList)),18,(35,random.choice(BlocksList)),18,(35,random.choice(BlocksList)),18,(35,random.choice(BlocksList))]
    row5 = [(35,random.choice(BlocksList)),18,18,18,18,18,18,(35,random.choice(BlocksList))]
    row6 = [0,0,0,17,0,0,0]
    row7 = [0,0,0,17,0,0,0]
    row8 = [0,0,0,(35,3),0,0,0]
    TempList = [row8,row7,row6,row5,row4,row3,row2,row1]
    return TempList

def blank():
    row1 = [0,0,0,0,0,0,0]
    row2 = [0,0,0,0,0,0,0]
    row3 = [0,0,0,0,0,0,0]
    row4 = [0,0,0,0,0,0,0]
    row5 = [0,0,0,0,0,0,0]
    row6 = [0,0,0,0,0,0,0]
    row7 = [0,0,0,0,0,0,0]
    row8 = [0,0,0,0,0,0,0]
    TempList = [row8,row7,row6,row5,row4,row3,row2,row1]
    return TempList


def PrintWall(ImportedList):

    pos = mc.player.getTilePos()
    mc.player.setPos(pos.x,pos.y,pos.z)

    myList = ImportedList
```

124

```
for row in range (0,8):
    for column in range (0,7):

        mc.setBlock(pos.x+column,pos.y+row,pos.z-20,myList[row][column])



while True:
    PrintWall(TreeBasic())
    time.sleep(2)
  # PrintWall(blank())
    PrintWall(TreeBasic())
    time.sleep(2)
```

Hack 4: Steve's Winter Wonderland by @b3ndavi3s

In this activity you will write a program that will help Steve get ready for Christmas

In your program you will need to	Open Minecraft Pi (menu - games- minecraft pi) and	Open Sonic Pi (menu - programming - sonic pi)

• program music • get the player's position • set a block • post messages • create loops • use delays to sequence the events • use variables	select Start Game Select an existing world or Create a New World Use the tab key to release the cursor from minecraft	You will need to use sonic pi 2.6 to get this type the following into the lx terminal - sudo apt-get update && sudo apt-get install sonic-pi Select a new workspace (buffer)

What are you trying to do?	Your pseudo-code

First you will need to program Sonic Pi to play Jingle Bells

use_bpm 120
2.times do
 2.times do
   play 71, release: 0.5
   sleep 0.5
 end
 play 71, release: 1
 sleep 1
end
	play 71, release: 0.5
	sleep 0.5
	play 74, release: 0.5
	sleep 0.5
	play 67, release: 0.75
	sleep 0.75
	play 69, release: 0.25
	sleep 0.25
	play 71, release: 2
	sleep 2

Now create a function called **jinglebells** by adding the following code around your music

define :jinglebells do

end

In another workspace (buffer) enter this code

in_thread do
 	
    loop do
    x, y, z = mc_location
	to execute more than one program
	creates a forever loop
	returns player's location

Code	Comment
| ```
        mc_set_area :snow_block , x-5, y-1, z-5, x+5, y-1, z+5
        2.times do
        mc_set_block :wood, x, y, z+5
        y=y+1
        end
        mc_set_area :leaves , x-3, y, z+5, x+3, y, z+5
        mc_set_area :leaves , x-2, y+1, z+5, x+2, y+1, z+5
        mc_set_area :leaves , x-1, y+2, z+5, x+1, y+2, z+5
        mc_set_block :leaves, x, y+3, z+5
        mc_set_block :snow, x, y+4, z+5
        end
 end

 in_thread do
        loop do
        mc_message "Ho Ho Ho!"
        sleep 4
        mc_message "Merry Christmas"
        sleep 10
        end
 end

 in_thread do
  loop do
        if one_in(2)
        use_synth :subpulse
        else use_synth :prophet
        jinglebells
        end
  end
 end
``` | Sets given area to snow<br>repeat twice<br>set wood block in given position<br>increase value of y by 1<br><br><br>sets given area to leaves<br><br><br><br>places snow on top of the tree<br>ends the forever loop<br>ends the thread<br><br><br><br>posts message to the screen<br>delay<br><br><br>ends the forever loop<br>ends the thread<br><br><br><br>creates probability of (0.5)<br>first condition<br>selects the synth<br>calls defined function<br>end selection statement<br>ends the forever loop<br>ends the thread |

Hack 5: Steve turns on the Christmas lights with pocket moneytronics Christmas tree / demos by @ncscomputing

Pictures of hack

Instructions

1. Download the code from : https://goo.gl/YOZyrB
2. Unzip and put on your pi.
3. Buy a pocket moneytronics Christmas tree from : http://www.pocketmoneytronics.co.uk/ / http://thepihut.com/products/gpio-xmas-tree-for-raspberry-pi
4. Follow instructions to put the hat on the gpio pins

5. Either open the 'TurningOnLightsWithSteve' file or create a new python file and type out the code for ' TurningOnLightsWithSteve' listed below:
6. Open minecraft and lay the blocks above
7. Press f5 to run, trouble shoot any errors
8. Go back to minecraft, right click on one of the blocks, sit back and "hacky christmas"

Code for the hack:

```
#Written by @ncscomputing / @warksraspijam


#import led controlling libraries # written by pocket moneytronics

import example_1 as e1

import example_2 as e2

import example_3 as e3

import example_4 as e4

import example_5 as e5

import example_6 as e6


#import minecraft libraries

import mcpi.minecraft as minecraft

import mcpi.block as block


mc = minecraft.Minecraft.create()


mc.postToChat("Welcome to Steves Warwickshire Raspberry Jam Christmas light turn on 2015 uisng pocket money
tronics christmas tree / and their 6 demo  programs :)")


while True:


  evs = mc.events.pollBlockHits()
```

```python
for e in evs:

    pos = e.pos


    b = mc.getBlock(pos.x,pos.y,pos.z)

    if b == 46:#tnt

        mc.postToChat("You chose Pocket Money tronics demo 1")

        e1.oneMain()

    elif b == 20:#glass

        mc.postToChat("You chose Pocket Money tronics demo 2")

        e2.twoMain()

    elif b == 41:#gold

        mc.postToChat("You chose Pocket Money tronics demo 3")

        e3.threeMain()

    elif b == 2:#grass

        mc.postToChat("You chose Pocket Money tronics demo 4")

        e4.fourMain()

    elif b == 26:#bed

        mc.postToChat("You chose Pocket Money tronics demo 5")

        e5.fiveMain()

    elif b == 45:#brick

        mc.postToChat("You chose Pocket Money tronics demo 6")

        e6.sixMain()
```
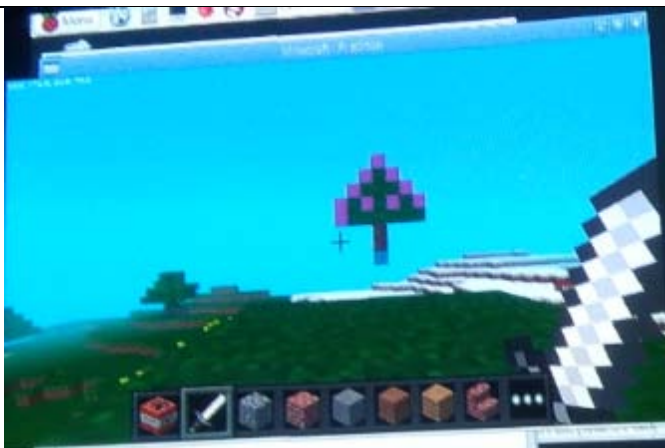
Hack 6: Incremental Christmas Tree by @ncscomputing

Lay a TNT block, then right click on it.	This will then output the wool id to screen and draw the Christmas Tree with wool lights. The colour of wool changes each time you right click.

Instructions

1. Download the zip file [here](#)

2. Unzip the folder on your Pi.

3. Open python and load the "TurningOnPIXELLightsWithSteve.py" file

4. OR create a blank python file and write the code below and save as above (TurningOnPIXELLightsWithSteve.py)

5. TurningOnPIXELLightsWithSteve.py imports the pixelated Christmas Tree as a library to make the code easier to understand

6. Open minecraft, create a new world

132

7. Lay a TNT block

8. Run the "TurningOnPIXELLightsWithSteve.py" file by pressing f5 or run module

9. Right click on the TNT block with a sword

10. Watch tree appear behind you

11. Right click again, see what happens

Code

```
#   by             /

import PixelChrisTree as pct# import tree

#import minecraft libraries
import mcpi.minecraft as minecraft
import mcpi.block as block

mc = minecraft.Minecraft.create()

mc.postToChat("Welcome to Steves Pixelated Christmas Tree :)")

woolId = 0
while True:

    evs = mc.events.pollBlockHits()
    for e in evs:
        pos = e.pos


        b = mc.getBlock(pos.x,pos.y,pos.z)
        if b == 46:#tnt
            woolId = woolId+1
            msg = "WoolID:",woolId
            mc.postToChat(msg)
            pct.christmasTreeMain(woolId)#white


        if woolId == 15:
            woolId = 0

```

Hack 7: Christmas messages with Minecraft Pi Microbit, hack by @ncscomputing mcpi microbit code by @whaleygeek, Minecraft Microbit built by @Pokemaster681

Pictures of the hack working

The default message is "happy chrimbo"	

Instructions

1. Download the original from here, or download the pi friendly files from here (https://goo.gl/5xp7GO)(the only difference is that I have adapted the csv and changed a few block ids that are compatible on Raspberry Pi)
2. Extract the folder and ensure it is saved on your Raspberry Pi.
3. Open Minecraft and create a new world.
4. Open the 'ChristmasMsg.py'.
5. Find the line that has the message "happy chrimbo".
6. Change to a message of your choice.
7. Go back to Minecraft, find an area that is fairly clear.
8. Go back to python and press f5 to run your code.
9. Then press space, and fly up and see the hack work and message displayed.

Extend yourself: try adding a mixture of messages or build a mini game that uses it as display board / scoreboard

Code for hack

```
File  Edit  Format  Run  Options  Windows  Help
import mc_microbit as m
import time

m.build()


while True:

    m.microbit.display.scroll("Happy Chrimbo")
    time.sleep(2)
```
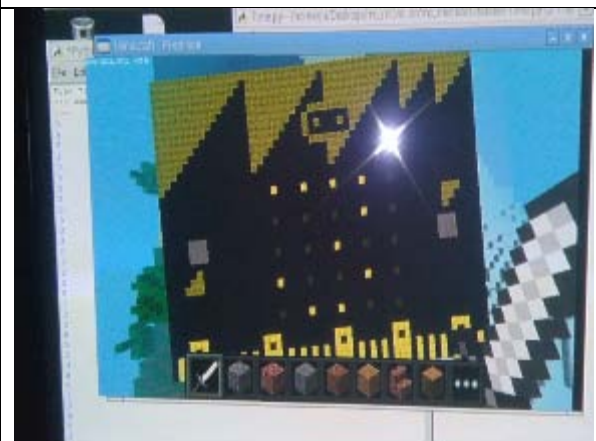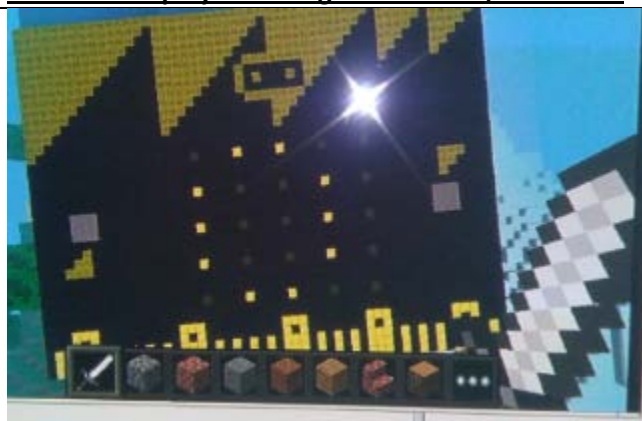
Hack 8: Minecraft Pi Microbit : Clock hack by @ncscomputing, MCPI Microbit code by @whaleygeek , Minecraft Microbit built by @Pokemaster681

Pictures of the hack working

The clock displays each digit one a time, see below	
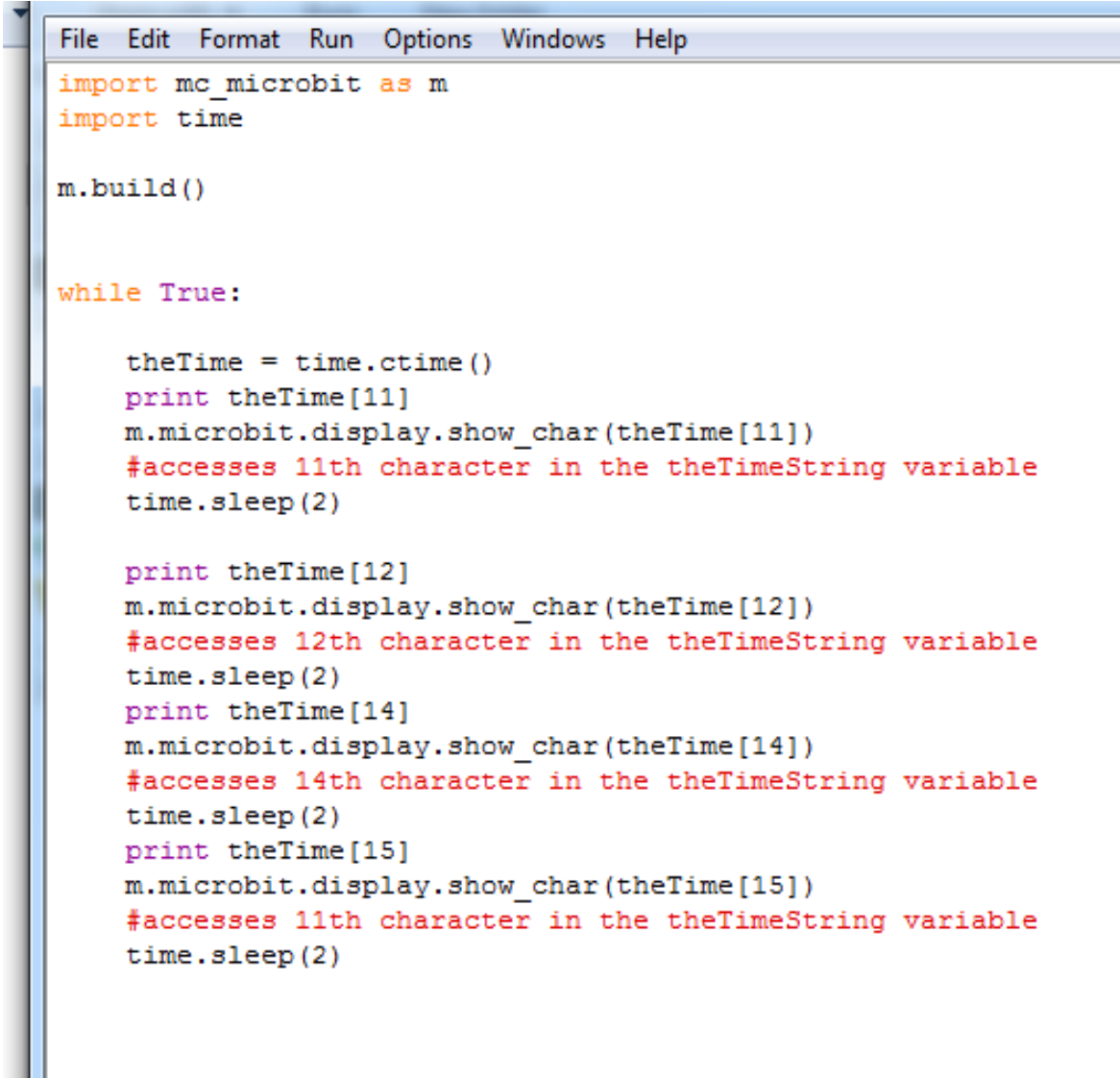
Instructions

10. Download the original from here, or download the pi friendly files from here (https://goo.gl/5xp7GO
1.)(the only difference is that I have adapted the csv and changed a few block ids that are compatible on Raspberry Pi)
2. Extract the folder and ensure it is saved on your Raspberry Pi.
3. Open Minecraft and create a new world.
4. Open the 'Clock.py' or click on the menu then go select programming and python 2 or 3 depending on your preference.
5. File > New document > and type in the Clock.py code listed below.
6. Save it as above.

7. Go back to Minecraft, find an area that is fairly clear.
8. Go back to python and press f5 to run your code, keep an eye on the shell for errors and then go back to minecraft and......
9. Then press space, and fly up and see the hack work and Time displayed. (Please note it will state the time your Pi displays, unless you are connected to the internet the time will be wrong as the Pi doesn't have an inbuilt clock)

Code for hack

```
File  Edit  Format  Run  Options  Windows  Help
import mc_microbit as m
import time

m.build()


while True:

    theTime = time.ctime()
    print theTime[11]
    m.microbit.display.show_char(theTime[11])
    #accesses 11th character in the theTimeString variable
    time.sleep(2)

    print theTime[12]
    m.microbit.display.show_char(theTime[12])
    #accesses 12th character in the theTimeString variable
    time.sleep(2)
    print theTime[14]
    m.microbit.display.show_char(theTime[14])
    #accesses 14th character in the theTimeString variable
    time.sleep(2)
    print theTime[15]
    m.microbit.display.show_char(theTime[15])
    #accesses 11th character in the theTimeString variable
    time.sleep(2)
```

Volume 4
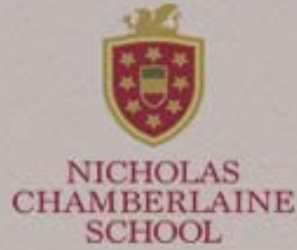
Hack 1: Hello World Java style @ncscomputing, @crazysqueak

Pic of hack working

N.b. this assumes that you have got Java working by working through hack 0, yet to be written up. I will do soon ☺, I have created this code in amongst the demos that Daniel Frisk created. I will refine this as and when time permits.

Instructions V1

1. Open up the Blue J IDE

2. Create a new class call it "Hello"

3. Type out the following code(below):

4. Then save the class file and compile it.

5. Now open up Minecraft and create a new world.

6. Close it down and right click on Hello box displayed on the BlueJ IDE. Right click on it and select the option displayed below :

new Test()

void main(String[] args)

Open Editor
Compile
Inspect
Remove

HelloWorld

s Edit Tools Options

ompile | Undo | Cut | Copy | Paste | F

```
package demo;

import pi.Minecraft.Events;
import pi.Minecraft;



public class HelloWorld {
    public static void main(String[] args) {
        Minecraft mc = Minecraft.connect(args);

        while (true) {

            mc.postToChat("Hello world");
        }

        }
}
```

To clarify the colon is the following type

:

;

The brackets used are:

}

7. Now click ok and say "hello" Java style.

Hack2: Modifying the Java Clock demo to print to post to chat original code by Dan Frisk tweaked by @ncscomputing
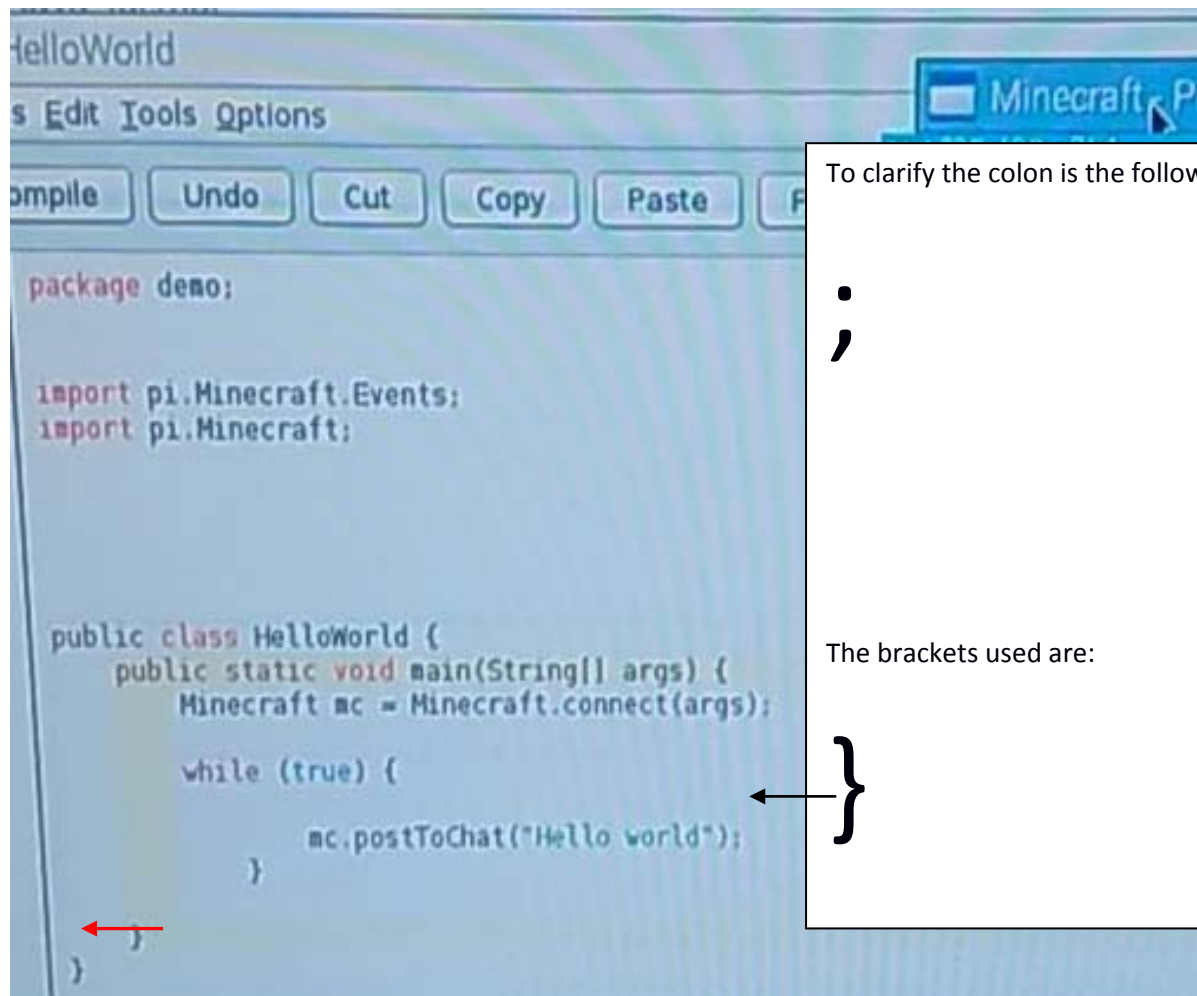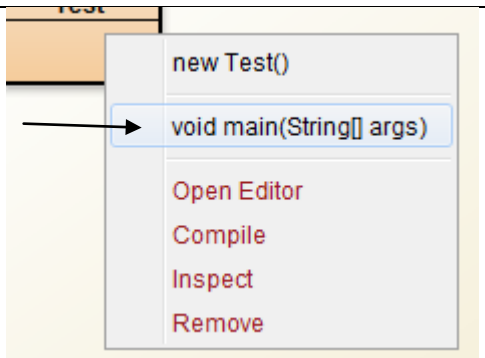
Pic of code working

N.b. this assumes that you have got Java working by working through hack 0, yet to be written up. I will do soon ☺, I have created this code in amongst the demos that Daniel Frisk created. I will refine this as and when time permits.
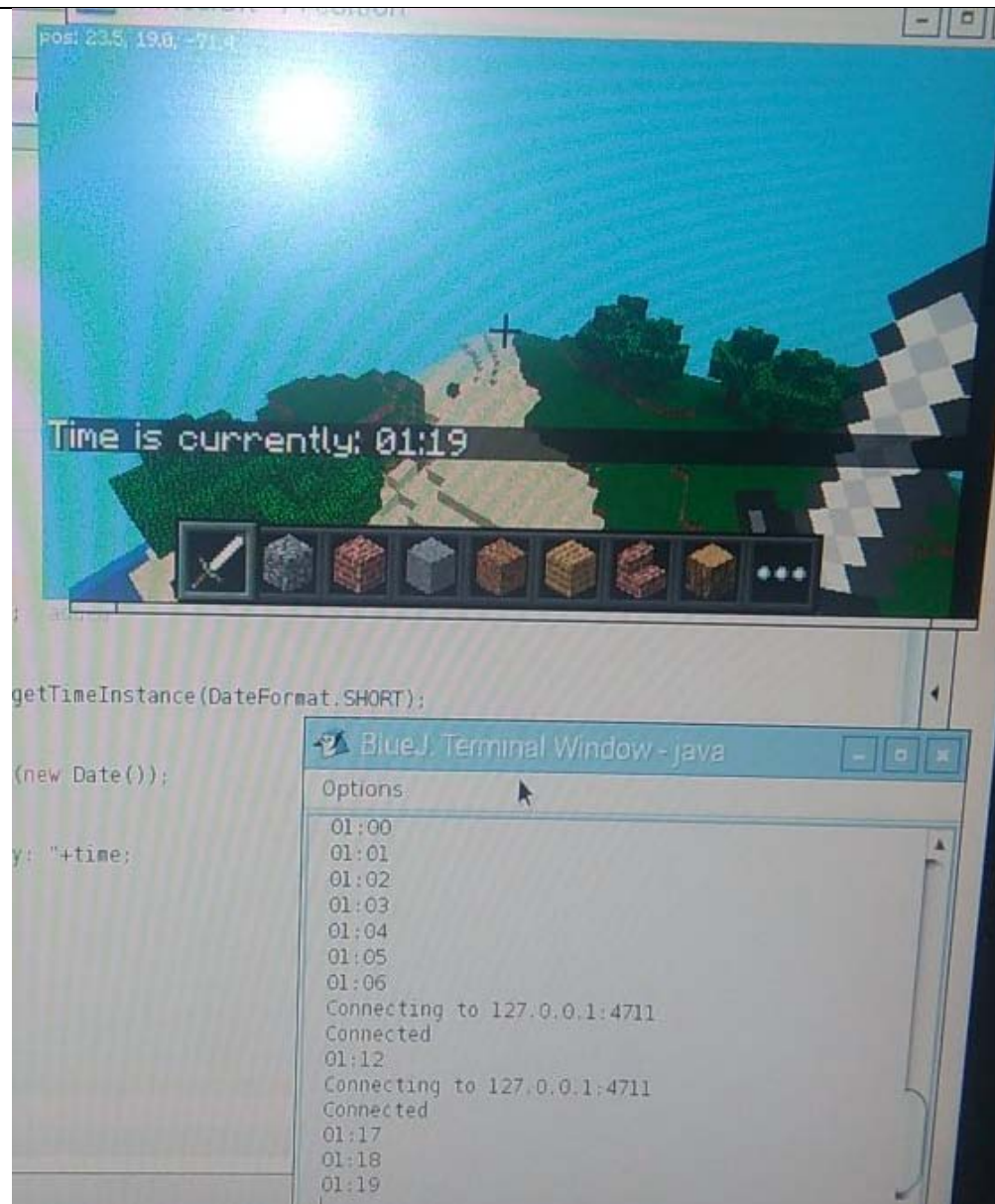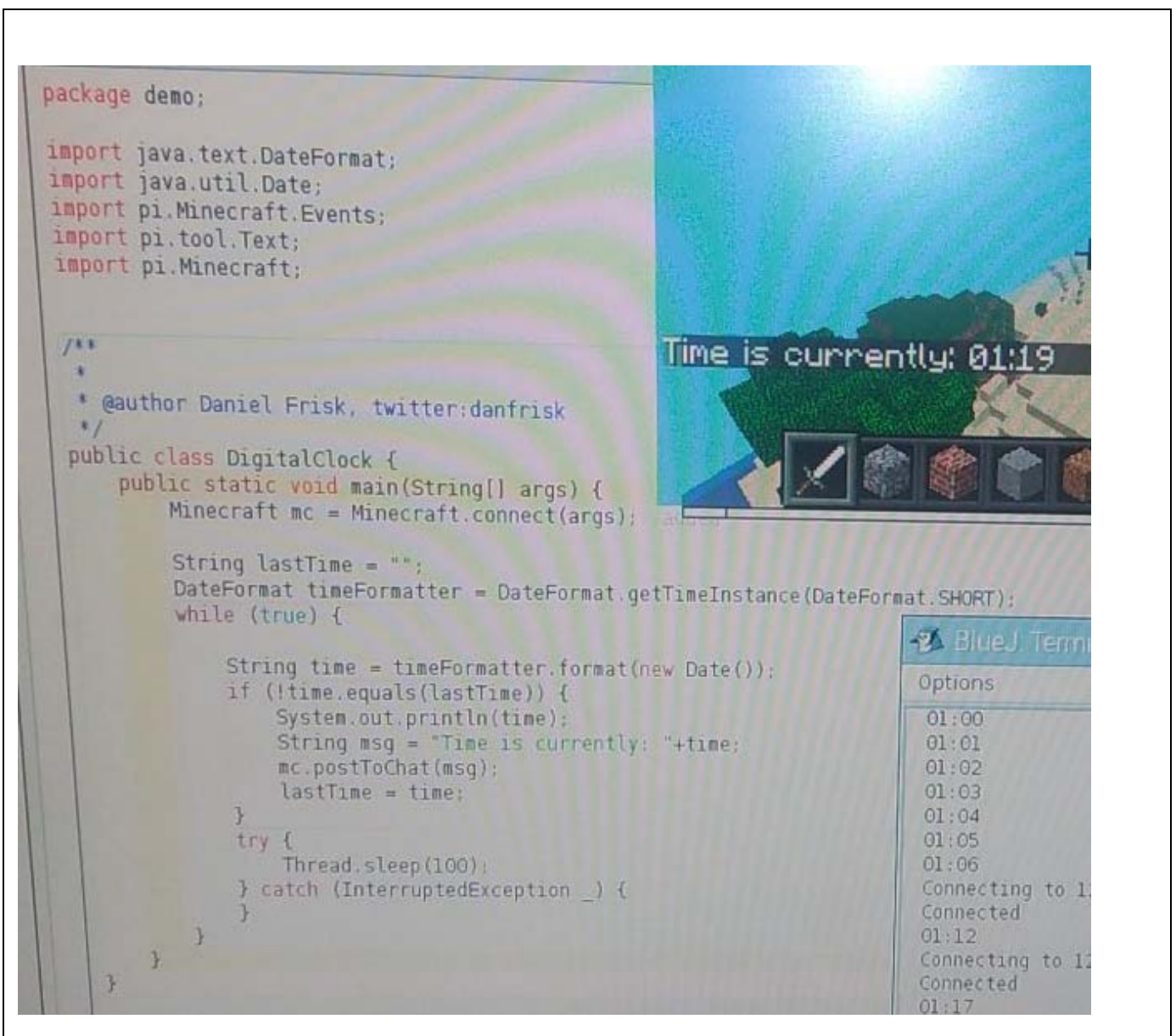
Instructions V1

The majority of this code is written by Daniel Frisk, it is included as part of the demo code which can be found in the following folder directory of your Raspberry Pi:

opt/minecraft-pi/add in when checked ☺

One of the demos is called DigitalClock, this is the code that I have added an additional few lines.

1. Open up the Blue J IDE with the Java Minecraft demos project imported.

2. Locate and open the DigitalClock class. Amend the existing code with the lines below.

```java
package demo;

import java.text.DateFormat;
import java.util.Date;
import pi.Minecraft.Events;
import pi.tool.Text;
import pi.Minecraft;


/**
 *
 * @author Daniel Frisk, twitter:danfrisk
 */
public class DigitalClock {
    public static void main(String[] args) {
        Minecraft mc = Minecraft.connect(args);

        String lastTime = "";
        DateFormat timeFormatter = DateFormat.getTimeInstance(DateFormat.SHORT);
        while (true) {

            String time = timeFormatter.format(new Date());
            if (!time.equals(lastTime)) {
                System.out.println(time);
                String msg = "Time is currently: "+time;
                mc.postToChat(msg);
                lastTime = time;
            }
            try {
                Thread.sleep(100);
            } catch (InterruptedException _) {
            }
        }
    }
}
```

3. Then save the class file and compile it.

4. Now open up Minecraft and create a new world.

5. Close the class file down and right click on DigitalClock box displayed on the BlueJ IDE. Right click on it and select the option displayed below :
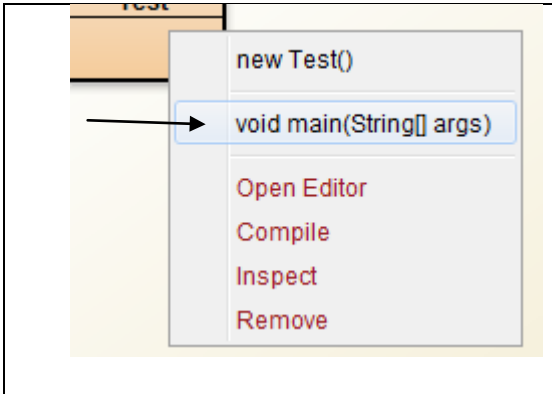
6. Click on ok then watch the time appear in the chat window every minute. If you are not connected to the internet the time may well be wrong ☺

Java hack 2 now done ☺ good work.

Hack 3: Coded Pixel Art: Storm trooper by @ncscomputing

Pictures of hack working (Think random colour wall, but much better ☺)

currently: 03:55

145

Instructions V2

For now the only instruction is:

1. Download it from [here](#)

2. Open it in Python 3

3. Open Minecraft and create a new world

4. Run it in python 3.

5. Read the function notes below to try and better understand how this code works.

Stretch yourself:

6. Change the block ids used in the storm trooper function call toward the bottom of the program

7. Duplicate the StarTrooper function and create your own pixel art can you create your own pixel art?

Function notes:

Function	Explanation
StarTrooper(B1,B2)	This function creates a 30 x 30 list of lists which contains two block ids B1 and B2.
	It prints to screen a storm trooper, with the colours you pass through as parameters
	It returns the list
	To print its contents to Minecraft use the Print wall function e.g. **PrintWall(RandomWall())**
RandomStormy(B1,BlocksList)	This function creates a 30 x 30 list of lists which contains two block ids B1 and a 2nd block ID which is randomly selected from the list at the beginning of the program.
	It returns the list
RandomWall()	This prints a 30 x 30 random colour wall
PrintWall(ImportedList)	Prints the random walls to screen in minecraft.
	You need to pass through the wall functions(those above) as parameter e.g. **PrintWall(RandomWall())** as they are return functions it is the equivalent of passing the list of lists straight through.

	e.g. of printing random storm trooper
	PrintWall(RandomStormy((35,random.choice(BlocksList)),BlocksList))

Hack 4 : GPIO zero whack a block, turn on a light by @ncscomputing

Picture of the hackworking
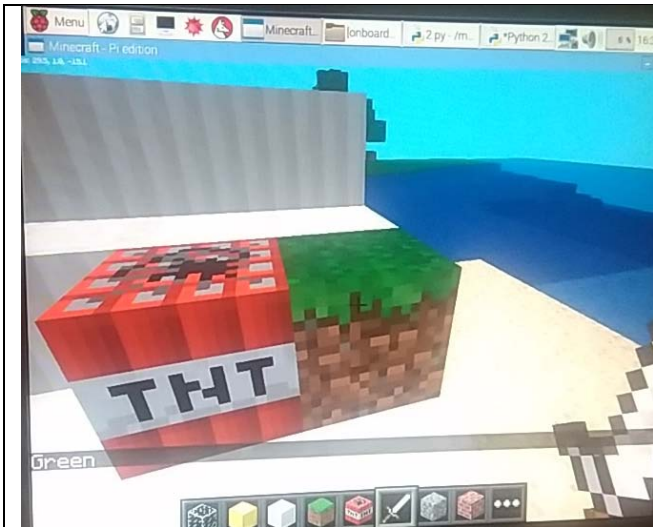
Whack grass and you turn on the green light on the "onboard Pi" green light

Whack tnt and you turn on the red light on the "onboard Pi" red light

The location of the lights are denoted above with the red rounded square

Instructions

1. This works using GPIO zero so please ensure that you have a recent updated version of Raspian Jessie OS.

2. Open python 3, create a version of the code listed below.

3. Save the python file, open minecraft and create lay the blocks shown in the above diagram.

4. Run the python code and right click on one of the two blocks. See what happens.

Code

```
#import minecraft libraries
import mcpi.minecraft as minecraft
import mcpi.block as block
```

```python
from gpiozero import LED
from time import sleep


mc = minecraft.Minecraft.create()

mc.postToChat("Turn on your Pi lights with MCPI and GPIO zero")

red = LED(35)
green = LED(47)

def RedOnMCPI(msg,time):
    mc.postToChat(msg)
    red.on()
    sleep(time)
    red.off()


def GreenOnMCPI(msg,time):
    mc.postToChat(msg)
    green.on()
    sleep(time)
    green.off()


while True:

    evs = mc.events.pollBlockHits()
    for e in evs:
        pos = e.pos

        b = mc.getBlock(pos.x,pos.y,pos.z)
        if b == 46:#tnt
            RedOnMCPI("Red",1)
        elif b == 2:#Grass
            GreenOnMCPI("Green",1)
```

Hack 5 : Whack a block: star wars special @ncscomputing featuring code from Martin O'Hanlon and Scott Turner

You can view video of this hack working here :

https://goo.gl/Y722ZT

Instructions

This code is a mix of work by Scott Turner and Martin O'Hanlon. I have added the 'pixelated storm trooper ' and 'whack a block' code on top to make it a sort of block based remote control. I have avoided putting the whole code in this booklet as it is too much. I think the focus should be on extending and editing it directly rather than coding from scratch.

Download the code here:
https://sites.google.com/a/nicholaschamberlaine.co.uk/minecrafthackingresources/hackpack-sow/whackablockstarwarsspecial.zip?attredirects=0&d=1
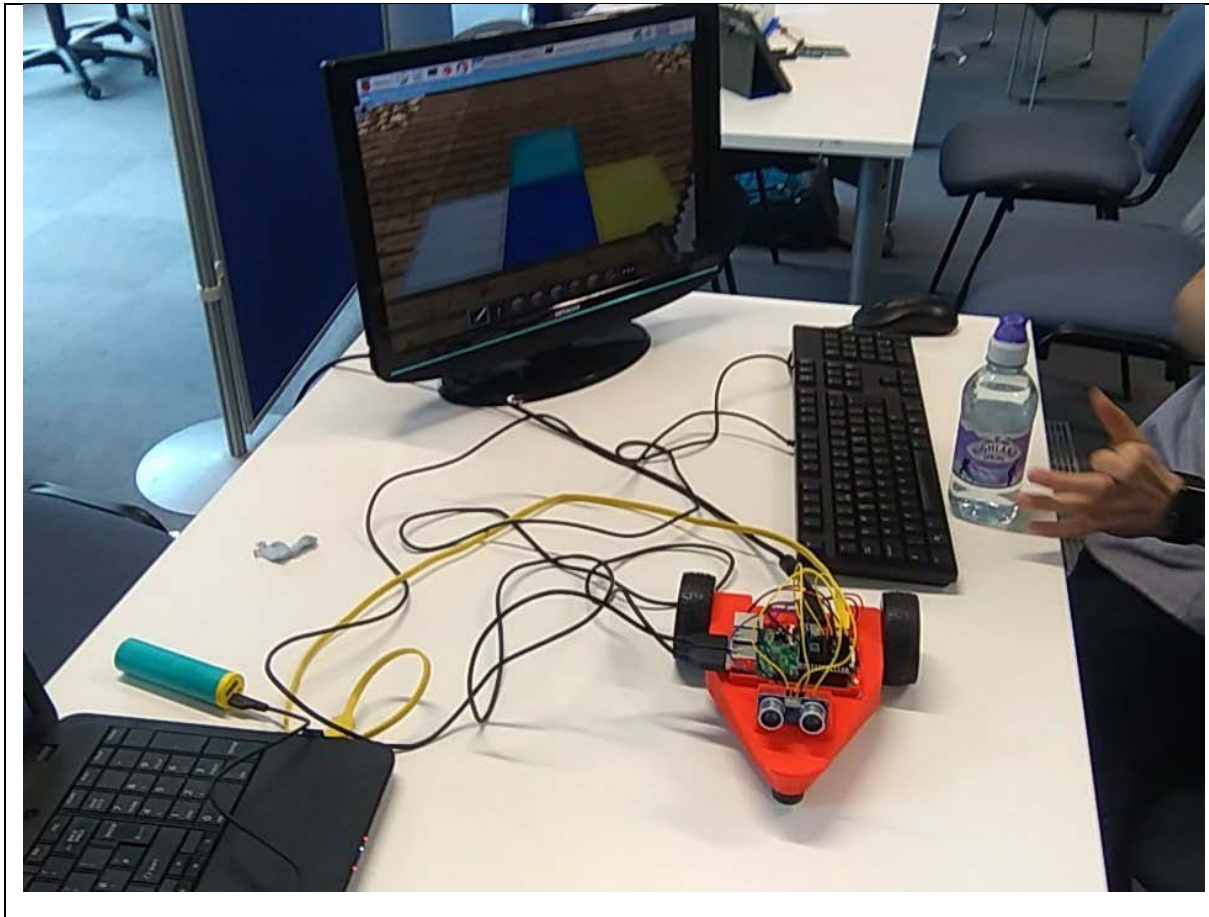
1. Extract the zip file

2. Open the whackablockstarwarsspecial.py file

3. Open minecraft pi and create a new world

4. Build a block based menu similar to the video example using the same blocks initially

5. Run the code.

Extension:

1. Adapt the code to run other python scripts when a block is whacked

2. Get it to display your own pixel art

3. Adapt the X wing into another design

Hack 6: Minecraft Controlled Robot(cam jam kit 3) CAM Jam kit team gpio code / NCS students mcpi code

Pic of hack working

Download code: https://sites.google.com/a/nicholaschamberlaine.co.uk/minecrafthackingresources/hackpack-sow/MCPI%20ROBOT.zip?attredirects=0&d=1

OR code yourself

Code 1 robot movement

```python
"""
+ Please note this code is taken from a student enrichment project.

+ They worked through the Cam Jam edu kit 3 tutorials, which can be downloaded from here:
http://camjam.me/?page_id=1035#worksheets

+ They said that they used this code to get the directions working. They have researched

... this off their own back.

+ The cam jam kit can be bought from here: http://thepihut.com/products/camjam-edukit-3-robotics

+ The stl file for the 3d printed chasis can be downloaded from here https://t.co/IFUesi0KrZ

+ This code is not written by me it was adapted my the kids who took part in the project

+ They are Reece, Connor, Adam and Harvey. They did a sterling job and deserve much credit :)


This script does the donkey work any way and is imported into the mcpi control script as a library.

This makes the mcpi implementation much cleaner.



#I have not written it. Cam Jam kit 3 might be the origin. Regardless it works.



I hope this is useful :) @ncscomputing / @warksraspijam


"""

#This python file will define the four directions

#The student that was in my school team found this code




import RPi.GPIO as GPIO # Imports the GPIO Library

import time # Imports the Time Library
```

153

```python
#Set the GPIO modes

GPIO.setmode(GPIO.BCM)

GPIO.setwarnings(False)


pinMotorAForwards = 10

pinMotorABackwards = 9

pinMotorBForwards = 8

pinMotorBBackwards = 7


GPIO.setup(pinMotorAForwards, GPIO.OUT)

GPIO.setup(pinMotorABackwards, GPIO.OUT)

GPIO.setup(pinMotorBForwards, GPIO.OUT)

GPIO.setup(pinMotorBBackwards, GPIO.OUT)


# Define StopMotors() which will turn off all motors

def StopMotors():

        GPIO.setmode(GPIO.BCM)

        GPIO.setwarnings(False)


        pinMotorAForwards = 10

        pinMotorABackwards = 9

        pinMotorBForwards = 8

        pinMotorBBackwards = 7
```

```python
        GPIO.setup(pinMotorAForwards, GPIO.OUT)

        GPIO.setup(pinMotorABackwards, GPIO.OUT)

        GPIO.setup(pinMotorBForwards, GPIO.OUT)

        GPIO.setup(pinMotorBBackwards, GPIO.OUT)


        GPIO.output(pinMotorAForwards, 0)

        GPIO.output(pinMotorABackwards, 0)

        GPIO.output(pinMotorBForwards, 0)

        GPIO.output(pinMotorBBackwards,0)


        GPIO.cleanup()


# Define Forwards() Which will turn the motors forwards

def Forwards():

        GPIO.setmode(GPIO.BCM)

        GPIO.setwarnings(False)


        pinMotorAForwards = 10

        pinMotorABackwards = 9

        pinMotorBForwards = 8

        pinMotorBBackwards = 7


        GPIO.setup(pinMotorAForwards, GPIO.OUT)

        GPIO.setup(pinMotorABackwards, GPIO.OUT)

        GPIO.setup(pinMotorBForwards, GPIO.OUT)

        GPIO.setup(pinMotorBBackwards, GPIO.OUT)
```

155

```python
        GPIO.output(pinMotorAForwards, 1)

        GPIO.output(pinMotorABackwards, 0)

        GPIO.output(pinMotorBForwards, 1)

        GPIO.output(pinMotorBBackwards,0)

        time.sleep(1)


        GPIO.cleanup()


#Define Backwards() which will turn the motors backwards

def Backwards():


        GPIO.setmode(GPIO.BCM)

        GPIO.setwarnings(False)


        pinMotorAForwards = 10

        pinMotorABackwards = 9

        pinMotorBForwards = 8

        pinMotorBBackwards = 7


        GPIO.setup(pinMotorAForwards, GPIO.OUT)

        GPIO.setup(pinMotorABackwards, GPIO.OUT)

        GPIO.setup(pinMotorBForwards, GPIO.OUT)

        GPIO.setup(pinMotorBBackwards, GPIO.OUT)
```

```python
        GPIO.output(pinMotorAForwards, 0)

        GPIO.output(pinMotorABackwards, 1)

        GPIO.output(pinMotorBForwards, 0)

        GPIO.output(pinMotorBBackwards,1)

        time.sleep(1)


        GPIO.cleanup()


# Define Turning Left

def Left():


        GPIO.setmode(GPIO.BCM)

        GPIO.setwarnings(False)


        pinMotorAForwards = 10

        pinMotorABackwards = 9

        pinMotorBForwards = 8

        pinMotorBBackwards = 7


        GPIO.setup(pinMotorAForwards, GPIO.OUT)

        GPIO.setup(pinMotorABackwards, GPIO.OUT)

        GPIO.setup(pinMotorBForwards, GPIO.OUT)

        GPIO.setup(pinMotorBBackwards, GPIO.OUT)



        GPIO.output(pinMotorAForwards, 0)
```

```python
        GPIO.output(pinMotorABackwards, 1)

        GPIO.output(pinMotorBForwards, 1)

        GPIO.output(pinMotorBBackwards,0)

        time.sleep(1)


        GPIO.cleanup()


# Define Turning Right

def Right():


        GPIO.setmode(GPIO.BCM)

        GPIO.setwarnings(False)


        pinMotorAForwards = 10

        pinMotorABackwards = 9

        pinMotorBForwards = 8

        pinMotorBBackwards = 7


        GPIO.setup(pinMotorAForwards, GPIO.OUT)

        GPIO.setup(pinMotorABackwards, GPIO.OUT)

        GPIO.setup(pinMotorBForwards, GPIO.OUT)

        GPIO.setup(pinMotorBBackwards, GPIO.OUT)


        GPIO.output(pinMotorAForwards, 1)

        GPIO.output(pinMotorABackwards, 0)

        GPIO.output(pinMotorBForwards, 0)
```

```
        GPIO.output(pinMotorBBackwards,1)


        time.sleep(1)


        GPIO.cleanup()


GPIO.cleanup()
```

Code 2 mcpi interaction

```
import RobotControls as rc

#This assumes that your python controller is called RobotControls.py

"""

inherited methods for reference purposes


rc.StopMotors()

rc.Forwards()

rc.Backwards()

rc.Left()

rc.Right()


"""


"""
```

+ Please note this code is taken from a student enrichment project.

+ They worked through the Cam Jam edu kit 3 tutorials, which can be downloaded from here: http://camjam.me/?page_id=1035#worksheets

+ They said that they used this code to get the directions working. They have researched

... this off their own back.

+ The cam jam kit can be bought from here: http://thepihut.com/products/camjam-edukit-3-robotics

+ The stl file for the 3d printed chasis can be downloaded from here https://t.co/IFUesi0KrZ

+ This code is not written by me it was adapted my the kids who took part in the project

+ They are Reece, Connor, Adam and Harvey. They did a sterling job and deserve much credit :)

This script does the donkey work any way and is imported into the mcpi control script as a library.

This makes the mcpi implementation much cleaner.

#I have not written it. Cam Jam kit 3 might be the origin. Regardless it works.

I hope this is useful :) @ncscomputing / @warksraspijam

+The script below is based from a tutorial that I wrote as a teaching resource at my school. Students have

-adapted this and imported the control code to hide the heavy work.

"""

#import minecraft libraries

import mcpi.minecraft as minecraft

import mcpi.block as block

mc = minecraft.Minecraft.create()

```python
mc.postToChat("Control the robot with steve ")


while True:


    evs = mc.events.pollBlockHits()

    for e in evs:

        pos = e.pos


        b = mc.getBlock(pos.x,pos.y,pos.z)

        if b == 46:#tnt = stop

            mc.postToChat("stop robot")

            rc.StopMotors()

        elif b == 42:#iron = left

            mc.postToChat("Left")

            rc.Left()

        elif b == 41:#gold = right

            mc.postToChat("Right")

            rc.Right()

        elif b == 57:#diamond = forward

            mc.postToChat("Forward")

            rc.Forwards()

        elif b == 22:#lapis = Backward

            mc.postToChat("Backward")

            rc.Backwards()
```

Pure Micro bit hacks with Micro python Hack 6: Hello world, hello Dave by @ncscomputing

This does the basic hello world and says a hello to a few more people along the way using a basic list data structure to store the names, try it out:

Key concepts used here are:

- Lists
- Iteration
- Index
- Count controlled loops
- Accessing values stored in a list
- joining together string values
- Casting

Code

```
import microbit

namesList = ["Dave","Fran","Beth"] #list of 3 names

index = 0 # count of current location in the loop / list

microbit.display.scroll("Hello World") # standard
while index <=2:
        msg = "Hello "+str(namesList[index]) # cherry on top :)
        microbit.display.scroll(msg)
        index = index +1
```

Pure Micro bit hacks with Micro python Hack 7: Using buttons, this is a test program by @ncscomputing

The second example uses the 'a' and 'b' buttons to print out messages to screen.

Testing out using the buttons try this code by creating a test program:

Code:

```
import microbit

while True:

    if microbit.button_a.is_pressed():
        microbit.display.scroll("This is a ...")

    if microbit.button_b.is_pressed():
        microbit.display.scroll("....test program")
```

Pure Micro bit hacks with Micro python Hack 8: Minecraft block id and description scroller by ncscomputing

This third program makes use of two lists and uses two lists to display some Minecraft block id's followed by the block name. It uses two lists and it prints out the values of each list after each loop/iteration through the list. Have a go a

Key concepts used here are:

- Multiple lists
- Iteration
- Index
- Infinite loops
- Accessing values stored in two lists
- Casting

Code

```
import microbit


BlockIdsList = [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19]


BlockNamesList = [

"Air",

"Stone",

"Grass",

"Dirt",
```

```
"Cobblestone",

"Wood Planks",

"Sapling",

"Bedrock",

"Water",

"Water Stationary",

"Lava flowing",

"Lava stationary",

"Sand",

"Gravel",

"Gold Ore",

"Iron Ore",

"Coal Ore",

"Wood",

"Leaves",]


Count = 0


while True:

    microbit.display.scroll(str(BlockIdsList[Count]))

    microbit.sleep(1000)

    microbit.display.scroll(str(BlockNamesList[Count]))

    Count = Count +1


    if Count == 19:

        Count = 0
```

Pure Micro bit hacks with Micro python Hack 9: Random nickname generator by @ncscomputing

The fourth example is a random nickname generator based on adjectives for body type and random names It uses the majority of the coding concepts previously looked at except it introduces the **random.choice function**

Code:

```
import microbit

import random

NameList = ["Dave","Frank","Dorothy","Roger","Tarquin","Simon","Melanie"]

BodyTypeAdjectiveList =
["Sturdy","Bullnecked","Gangling","Heavyset","Lanky","Musclebound"]
while True:
    RandomNickname = random.choice(BodyTypeAdjectiveList)+" "+random.choice(NameList)
    microbit.display.scroll(RandomNickname)
```

Dice roller

This simple dice simulation introduces the accelerometer and how to use it to randomly simulate a 6 sided dice.

Code

```python
from microbit import *
import random

DiceNumbers = [1,2,3,4,5,6]#List of 6 possible numbers

while True:
    if accelerometer.was_gesture('shake'):# if shaken then

        """create a string variable which puts out an intro message and joins it
        with a random number from the DiceNumbers list
        #cast the number to string so it can be displayed
        """

        msg = "You rolled.. "+str(random.choice(DiceNumbers))
        display.scroll(msg)#show the text
```

Shake the bit, display a random picture

This basically allows you to shake the microbit and this will randomly display one of the library of images. N.B. I have only implemented a few to give a brief idea :)

Concepts covered:

- Lists
- Functions
- Loops
- Accelerometer
- Random library *Conditional statements

Code

```
from microbit import *
import random

"""
List of  possible pictures not sure if it is exhaustive, stored as string so they can
be stored in a list
this process is called casting.
"""
PicNamesList =
[str(Image.SAD),str(Image.HEART),str(Image.MEH),str(Image.RABBIT),str(Image.COW)]
```

```
#other possible images I've not included as it is OTT for an example :)
#Image.DUCK, Image.SWORD, Image.GIRAFFE, Image.TARGET, Image.HOUSE, Image.TORTOISE,
Image.UMBRELLA,
#Image.SNAKE, Image.SKULL, Image.BUTTERFLY, Image.TSHIRT, Image.XMAS,
Image.STICKFIGURE,
#Image.MUSIC_CROTCHET, Image.CHESSBOARD, Image.YES, Image.NO, Image.CONFUSED,
Image.ANGRY,
#Image.ARROW_N, Image.ARROW_S, Image.ARROW_E, Image.ARROW_W, Image.ARROW_SE]


"""
I have created a function that groups the code and makes the final program much cleaner
it basically:
*imports the list as a parameter
*creates a temp variable which stores the string representation of the image randomly
selected
*then uses if and elif statements to check which image it should display on microbit
"""

def checkWhichImageIAm(PicNamesList):
    chosenImage = random.choice(PicNamesList)
    if chosenImage == str(Image.SAD):
        display.show(Image.SAD)
    elif chosenImage == str(Image.HEART):
        display.show(Image.HEART)
    elif chosenImage == str(Image.MEH):
        display.show(Image.MEH)
    elif chosenImage == str(Image.RABBIT):
        display.show(Image.RABBIT)
    elif chosenImage == str(Image.COW):
        display.show(Image.COW)
    #you can implement every image if you want to....


while True:
    if accelerometer.was_gesture('shake'):# if shaken then
        checkWhichImageIAm(PicNamesList)# run check which image am I function
```
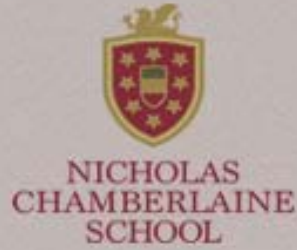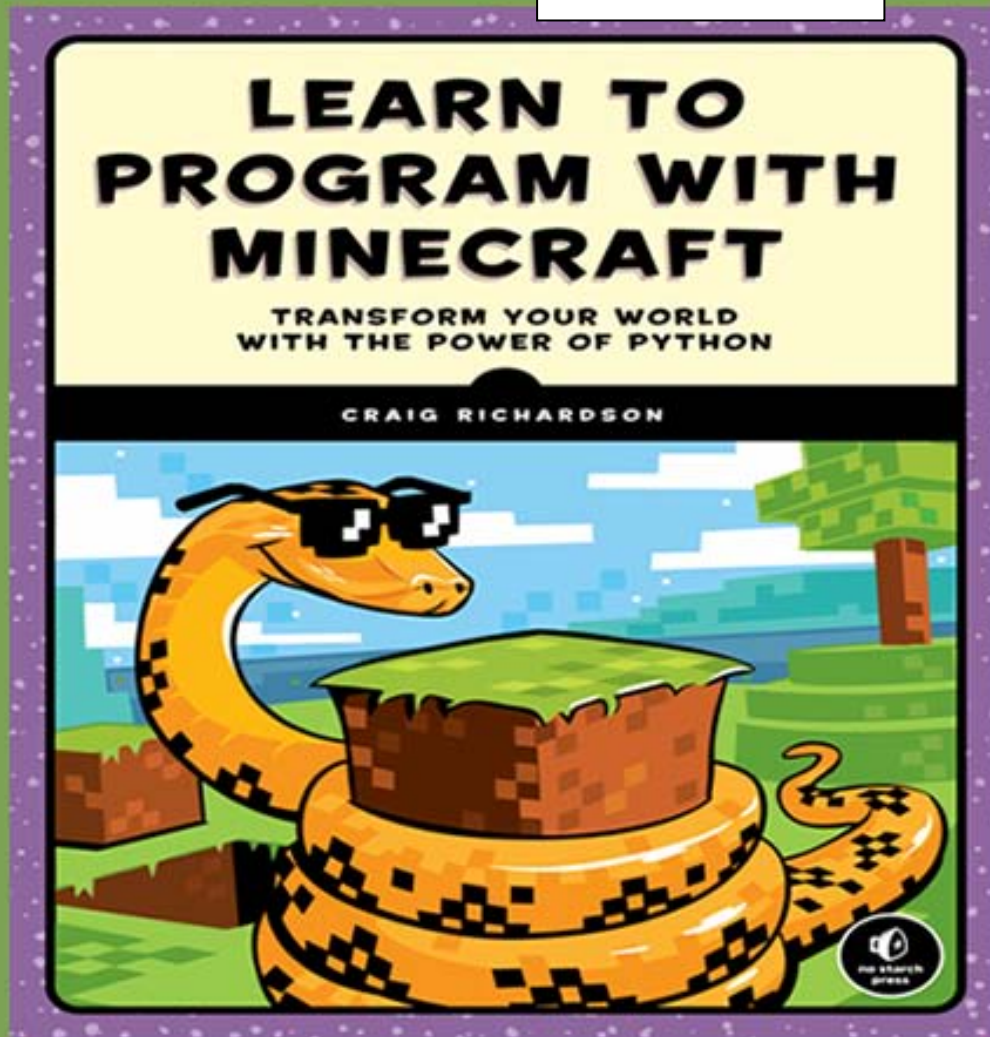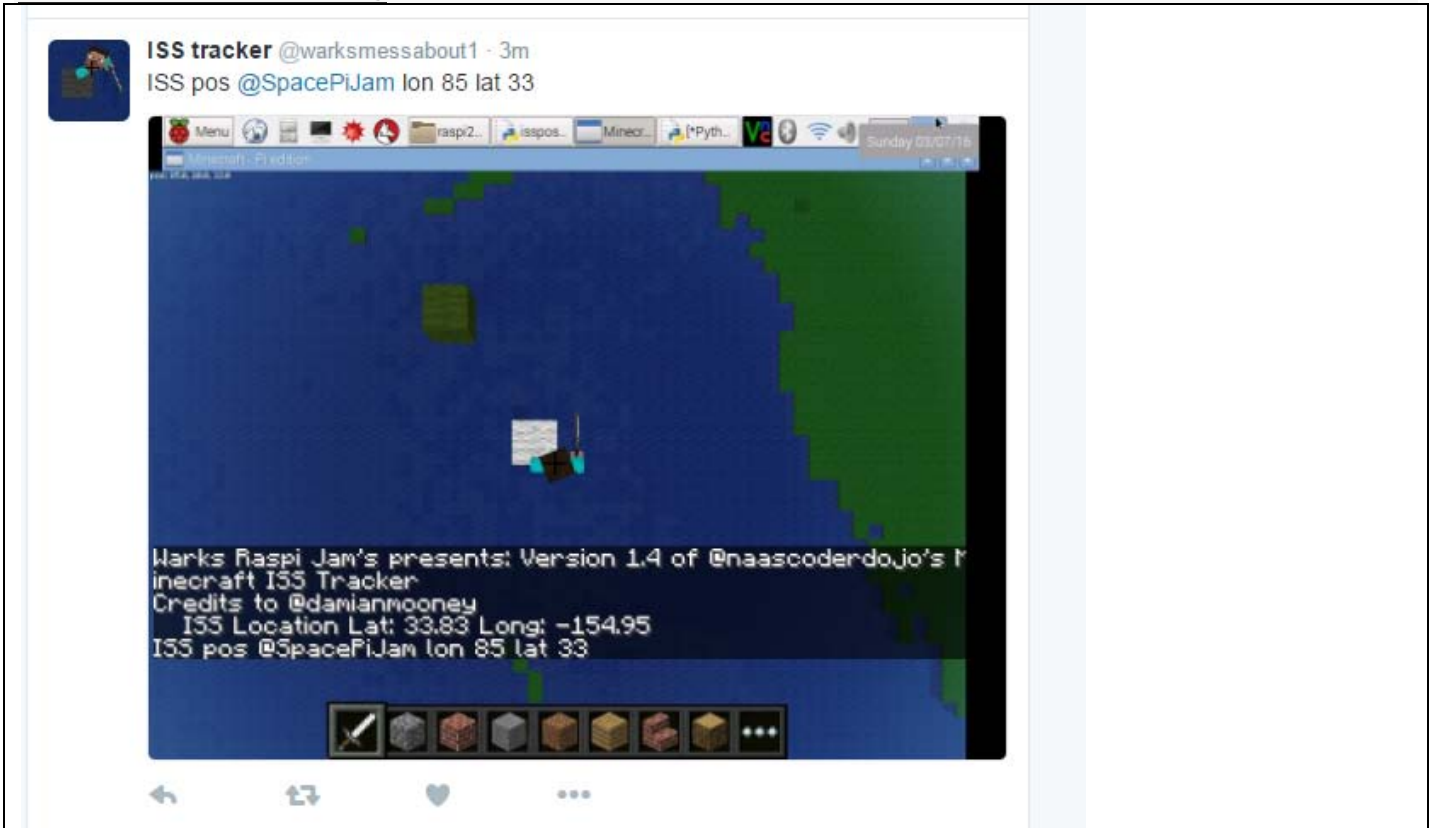
Volume 5

NICHOLAS
CHAMBERLAINE
SCHOOL

RASPBERRY JAM
★ WARWICKSHIRE ★

MINECRAFT
RASPBERRY JAM
★ WARWICKSHIRE ★

Hack pack Volume 5
"LTPWMC remixes"

and much more.............

LEARN TO PROGRAM WITH MINECRAFT

TRANSFORM YOUR WORLD WITH THE POWER OF PYTHON

CRAIG RICHARDSON

Hack 1: Tweeting Minecraft ISS tracker @ncscomputing @damianmooney

`Picture of hack working`

Instructions

1. Download Damian Mooney's map rendering script from here you will also need the ascii text file that it reads from here

2. Open the script in python2 Idle . Save the text file in the same folder or else it will not be able to read it.

3. Open minecraft, create a new world and run the script to create your map.

4. Damian's map is awesome!

5. Now to get your bot tweeting you will need to do a few things.

   a. Install raspi2png via Martin O'Hanlons tutorial (here)

   b. Install twython python library and create a twitter app which is covered in the amazing Raspberry Pi foundation tutorial for tweeting babbage available (here) This is what I used so I know that it works.

     c.   Ensure that the Python script for the twitter bot that you are about to make is saved in the same folder as Raspi2png or else it will not work. (To do with the sub process shell script code)

6.   Now here is the code for the ISS Twitter bot go and explore ☺

<u>Code:</u>

```
"""

Tweeting ISS current position at set intervals with Minecraft Pi.


Get current ISS position from http://wheretheiss.at/ and map it on

a raspberry pi with minecraft


Damian Mooney wrote this minecraft Tracker in 2016. I have added a few features on top, namely:


walking random wool blocks to mark current and positions

ability to screenshot the position using Martin O'Hanlons raspi2png tutorial:
(http://www.stuffaboutcode.com/2016/03/raspberry-pi-take-screenshot-of.html)

ability to tweet that using twython python library


"""

__author__ = '@damianmooney' additions by @ncscomputing

from mcpi import minecraft as minecraft

from mcpi import block as block

from datetime import datetime

import time

import urllib2

import json
```

```python
import random

import subprocess # to run shell script to autocall raspi2png

import sys

from twython import Twython


consumer_key = ''#goes here

consumer_secret = ''goes here

access_token = ''#goes here

access_token_secret = ''#goes here


api = Twython(consumer_key,consumer_secret,access_token,access_token_secret)


WoolList = [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15]


def getiss():
    """ call where the iss at api thanks to Bill Shupp"""
    response = urllib2.urlopen('https://api.wheretheiss.at/v1/satellites/25544')
    mydata = response.read()
    return mydata
```

```python
def do_coord(longitude):
    """ longitude: convert our longitude to a minecraft co-ordinate"""
    mine_long = longitude * -.55
    return mine_long




if __name__ == "__main__":
    mc = minecraft.Minecraft.create()
    mc.postToChat("   Minecraft ISS Tracker for @naascoderdojo")
    mc.player.setting("autojump", False)
    mc.player.setPos(6, 20, 50)
    while True:
        iss = getiss()
        pos = json.loads(iss)
        lat = pos['latitude']
        lon = pos['longitude']
        mc.postToChat("Warks Raspi Jam's presents: Version 1.4 of @naascoderdojo's Minecraft ISS Tracker")
        mc.camera.setFollow()
        mc.postToChat("Credits to @damianmooney")
        mc.postToChat('   ISS Location Lat: %.2f Long: %.2f' % (lat,lon))
        new_long = do_coord(lon)
        mc.player.setPos(int(new_long),  20, int(lat))
        mc.setBlock(int(new_long),  20-1, int(lat),35,random.choice(WoolList))
        msg = 'ISS pos '+'@SpacePiJam'+' lon %d lat %d' % (new_long, lat)
        print msg
        mc.postToChat(msg)
```

```
    a=subprocess.check_output('./raspi2png -d 3 -p "myscreenshot.png"',shell=True)

    photo = open('/home/pi/raspi2png/myscreenshot.png', 'rb')#change the directory to location that you have
saved raspi2png in

    response = api.upload_media(media=photo)

    api.update_status(status = msg, media_ids=[response['media_id']])

    time.sleep(600)  # --only update once every 10 minute
```

Hack 3:

```
from gpiozero import CamJamKitRobot


rc = CamJamKitRobot()


#This assumes that your python controller is called RobotControls.py
"""
inherited methods for reference purposes


rc.StopMotors()

rc.Forwards()

rc.Backwards()

rc.Left()

rc.Right()
"""
from twython import TwythonStreamer

#import InteractiveStormy as intStormy
```

177

```
import keyinfo as ki


class TweetStreamer(TwythonStreamer):
  def on_success(self, data):
    consumer_key = ki.ck()

    consumer_secret = ki.cs()

    access_token = ki.at()

    access_token_secret = ki.ats()


    left = "left"

    right = "right"

    forward = "forward"

    back = "back"


    if 'text' in data:
      tweet = data['text'].encode('utf-8')

      print tweet

      Components = tweet.split()

      #print Components[0]


      Count = 0

      while Count < len(Components):


        if Components[Count] == left:

          print left
```

```python
                rc.left()

            elif Components[Count] == right:

                print right

                rc.right()
            elif Components[Count] == forward:

                print forward

                rc.forward()

            elif Components[Count] == back:

                print back

                rc.backward()
            elif Components[Count] == "stop":

                print "stop"

                rc.stop()

            Count = Count +1

    def on_error(self, status_code, data):

        print status_code

        self.disconnect()


consumer_key = ki.ck()

consumer_secret = ki.cs()
```

```
access_token = ki.at()

access_token_secret = ki.ats()


streamer = TweetStreamer(consumer_key, consumer_secret,access_token, access_token_secret)


streamer.statuses.filter(track = '@warksmessabout1')
```

Hack 2: Minecraft controlled robot: Gpio zero @ncscomputing

Picture of robot:

180
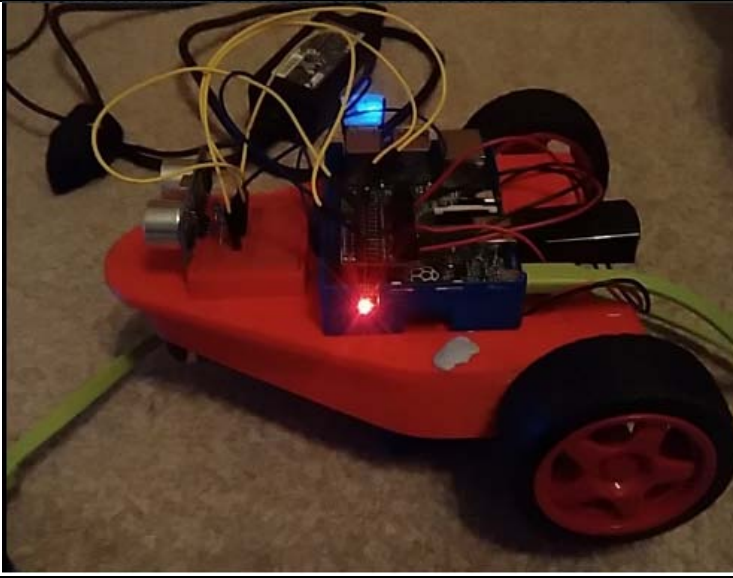
Instructions:

1. Ensure you have the latest version of Raspian Jessie or update to ensure you have the latest version of GPIO zero installed by following the instructions below:

Install

First, update your repositories list:

```
sudo apt-get update
```

Then install the package of your choice. Both Python 3 and Python 2 are supported. Python 3 is recommended:

```
sudo apt-get install python3-gpiozero
```

or:

```
sudo apt-get install python-gpiozero
```

Source : https://gpiozero.readthedocs.io/en/v1.2.0/

2. This assumes that you have a Cam Jam Edu kit 3 robot built with the wheels / motors connected. You do not need to have any of the sensors attached for it to work.

3. Open Python 3, create a new window and type the following code.

4. Save the file, open minecraft, create a D pad shape of the following blocks:

a. (left = bookcase)

b. (right = gold)

c. (stop aka centre = diamond)

d. (back = glass)

e. (forward = grass)

5. Minimise minecraft, run the Python code and correct any errors, go back to minecraft and whack one of the dpad blocks by right clicking. Well done !:)

Code:

```
from gpiozero import CamJamKitRobot

import time


rc = CamJamKitRobot()




#import minecraft libraries

import mcpi.minecraft as minecraft

import mcpi.block as block




mc = minecraft.Minecraft.create()




mc.postToChat("Control the robot with steve ")




while True:


  evs = mc.events.pollBlockHits()
```

```
for e in evs:

  pos = e.pos


  b = mc.getBlock(pos.x,pos.y,pos.z)

  if b == 57:#diamond = stop

    mc.postToChat("stop robot")

    time.sleep(1)

    rc.stop()

  elif b == 47:#bookcase = left

    mc.postToChat("Left")

    rc.left()

    time.sleep(1)

  elif b == 41:#gold = right

    mc.postToChat("Right")

    rc.right()

    time.sleep(1)

  elif b == 2:#grass = forward

    mc.postToChat("Forward")

    rc.forward()

    time.sleep(1)

  elif b == 20:#glass = Backward

    mc.postToChat("Backward")

    rc.backward()

    time.sleep(1)



for e in evs:
```

Learning to program with Minecraft Remix 1 Big wool cube: here today and gone in ten seconds by @ncscomputing

Picture of hack working

This is an adapted version of the original idea from the above book, which can be found on page 158 mission #44. It basically prompts the user for which colour of wool they want their block to be between 1 and 16. Then the size of the block and then it builds it and then clears it 10 seconds later.
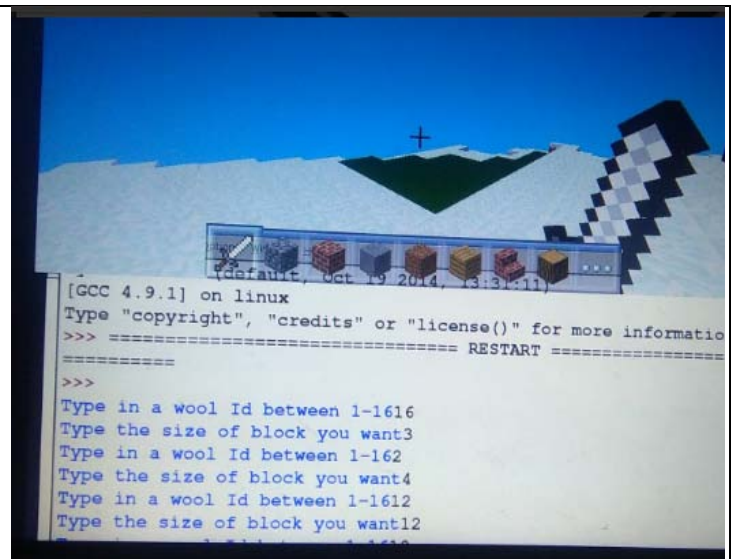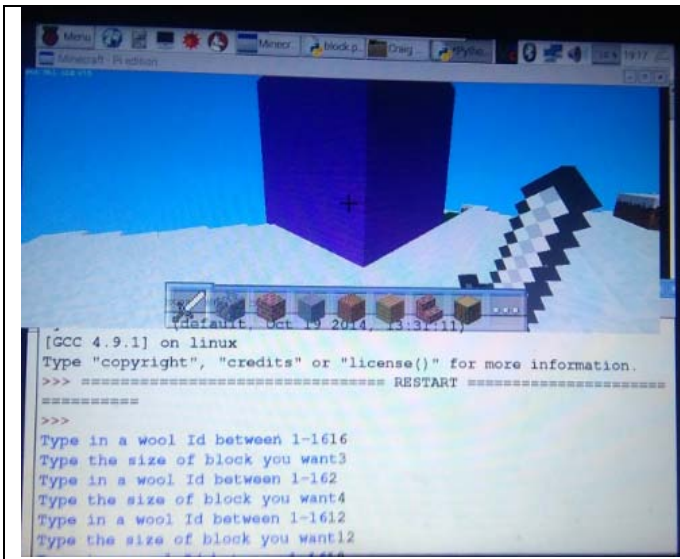
Code

```
from mcpi.minecraft import Minecraft

import time

mc = Minecraft.create()

block = 35

state = 0


def bigBlock(size,woolcolour):

    mc.setBlocks(pos.x, pos.y, pos.z,pos.x+size, pos.y+size, pos.z+size, 35,woolcolour)


def bigBlockClear(size,blockid):

    mc.setBlocks(pos.x, pos.y, pos.z,pos.x+size, pos.y+size, pos.z+size, blockid)


pos = mc.player.getTilePos()
```

185

```
while True:

    woolcolour = int(input("Type in a wool Id between 1-16"))

    size = int(input("Type the size of block you want"))

    bigBlock(size,woolcolour)

    time.sleep(10)

    bigBlockClear(size,0)
```
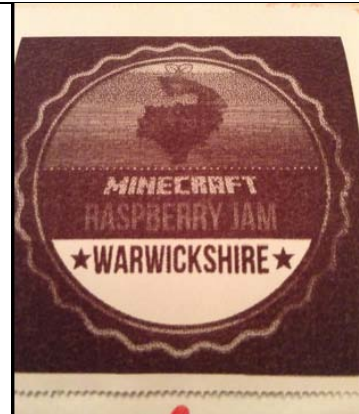
Adverts as a small way of saying thanks to those companies / people who have had an input / helping hand

You can find out more info at: http://www.pipsta.co.uk/		

186

THE LITTLE PRINTER
WITH BIG IDEAS

Compact, easy to set up and use Mini-Printer with NFC capabilities.

Print from Scratch or Python to produce QR codes, stickers, certificates, banners, coupons, puzzles – in fact anything you can think of!

Complete with National Curriculum orientated applications that work out-of-the-box and comprehensive tutorials that allow the system to meet specific needs, this Cloud capable package is the perfect accessory for the "Internet of Things" revolution.

Features

- Transparent, self-assembly, laser cut acrylic case with access to all connectors on your Pi
- USB interface for fast, convenient setup and fast transmission
- NFC interface for printing and secure authentication with NFC-enabled Android devices
- Mains power adaptor for faster printer
- Print to paper rolls or labels
- Comprehensive tutorials will guide you all the way from assembling the unit to printing
- Full instructions on setting up your own "Internet of Things" Cloud printer
- All the code required to print from Scratch and Python

Kit Contents*

- Pipsta USB and NFC Thermal Mini-Printer
- Self-assembly acrylic housing and all fixings
- Self-adhesive, non-slip feet
- 7.5Vdc Universal Mains power adaptor
- USB 'A' to Mini 'B' lead to connect Pi and Printer
- 1x paper and 1 x label roll to get you printing right away

Specification

	Pipsta case
Use with	Optimised for Raspberry Pi Model B+ (also suitable for use with A+)
Dimensions (mm)	85H x 95W x 124L
Housing Materials	Clear 3mm Acrylic
Connector Access	USB Ethernet (B+ only) HDMI RCA Video Audio Jack Micro SD Card

	Pipsta Printer
Model	Able AP1400 Thermal Panel Printer
Method	Direct Thermal
Media Types	Thermal Paper and Label rolls
Media Width	58mm
Print Width	48mm
Dots	384 dots per line, at 8 dots/mm
Roll Diameter	36mm
Communications Interfaces	USB Dynamic NFC RS232
Print Speed	Up to 80mm/s
Print Capabilities	Text, graphics, barcodes
Power Supply	4.8-9.0Vdc, 2A typical
Reliability - MCBF	10_8 print pulses or >50km. >13 million print lines (approx. 3,500 paper rolls.)

Wiley Adventures in Minecraft