**Code conversation level 1- your biggest fan**

**Describe what the whole program is doing in a couple of sentences:**

**Fill in the blanks**

|  |  |
| --- | --- |
| ***Code*** | ***What is this line doing?*** |
| from microbit import \* | imports the micro:bit module to give you access to all the hardware that is built-in into your board |
| Initialising the variables |  |
| min\_power = 600 | set value of min\_power variable to 600 |
| max\_power = 1023 |  |
| power\_step = (max\_power - min\_power) / 9 | set power\_step variable to be equal to max\_power minus min\_power divided by 9 |
| speed = 0 |  |
| This function will set the speed of the fan |  |
| def set\_power(speed): | define a micro:bit function called set\_power () |
| display.show(str(speed)) | call the micro:bit display.show() function which displays the variable speed on the micro:bit LED screen. It needs the str to convert the variable speed to a string (i.e. text not number) as the display.show displays only strings. |
| if speed == 0: |  |
| pin0.write\_analog(0) | ...call the inbuilt function and pass the value 0 |
| else: | otherwise… |
| pin0.write\_analog(speed \* power\_step + min\_power) | ...call the inbuilt function pin0.write\_analog() to calculate and pass the speed value via pin0 |
| set\_power(speed) | call the function set\_power and passes the parameter speed to it |
|  |  |
| while True: |  |
| if button\_a.was\_pressed(): | if button a is pressed then… |
| speed -= 1 | …take 1 off the variable speed (this is a short-hand way of saying speed = speed -1) |
| if speed < 0 : |  |
| speed = 0 |  |
| set\_power(speed) | call set\_power() function and pass parameter speed to set the appropriate power level of the fan |
| elif button\_b.was\_pressed(): | …otherwise if button b is pressed then... |
| speed += 1 |  |
| if speed > 9: |  |
| speed = 9 |  |
| set\_power(speed) | call set\_power() function and pass parameter speed |
| sleep(100) | put a pause of 100 milliseconds in the program |