1. I believe that my program is correct because the only place for an error occurs when user is entering something in. Based on the told assumption of only numbers. By checking 1 >= x <=9 we catch any and all errors. However, if a letter or special character is entered it would cause the program to break. On top of testing I would like to have someone non-comp sci oriented try to use it and see if it breaks. I think there is almost no chance of it doing something I do not expect
2. There are some aspects that are not fully efficient. Using the concept of iteration over a list I now realise for example using a for loop to check one of the diagonals is not needed and is slower. There is a good reason for this (see next answer) however, the rest of the program is quite boiled down. Any efficiency increase would have come at understandability cost which I thought is not worth it.
3. Because I use very few hard-coded values making the program deal with a larger square and bigger sum would be really easy. However, the only place where I cheeped-out is checking the up-right diagonal. Some places I used range(3) instead of range(len(square)) but that can be changed without shaking up any other parts of the code.
4. I think my program is really robust. Because when the range is checked any failure to meet criteria results in a false, which triggers a print(“NO”). meaning as long as it is a number the program will always work and just print NO. However, any letter makes it crash.
5. I think it took a total of 7 hours of work. Including time not spent coding but thinking (while walking or eating lunch). However, some of this time was done during another lecture (1h20min). About 4 of these hours was in one sitting meaning the return on time started to diminish. I thought it would take 2-3 hours, but the testing took longer as it required tedious copy pasting inputs, output and reason into python. (Looking back on it, there probably was a faster way). Also, wrote in the wrong order once and thus had to change functions around. Nothing really surprised me as the design outline basically gave away the answer.
6. I found the best work I did was in quite. Sometimes however, working in the lab meant a lot of noise and thus the need for music. Which I believe tires the brain faster. I have decided to work 1 or 2 hours a day at home. If by Thursday I still help I can go shortly to Mike but hours in the evening is too much.