An arithmetic problem:

Imagine the following situation. The white queen captures a black pawn diagonally that is a few squares away. First, the queen is moved towards the pawn, but stops one square before it reaches it. We will call this the ‘rest square’ Second, the pawn is being moved to a graveyard square. Third, the queen moves to the place where the pawn stood.

The first and second part are inside the trash() function. The third part is in the zet() function. That is necessary, because the zet() function always runs, and it needs to do something. The queen should, however, not move from (x1,y1) to (x2,y2) but from the rest square to (x2,y2) in this third part.

Because we want to keep the Redo function intact, we don’t want to change (x1,y1) in the trash function. We can’t add local variables (eg. (x4,y4) to save the coordinates of the rest square, because these variables would be useless in the zet() function. We don’t want extra global variables because we have too many already. Yet still it needs to know where the rest square is. Now, how do we accomplish this?

The queen first stands on (x1,y1). The pawn stands on (x2,y2). For your convenience, we will only look at the x-direction; the calculations for the y-directions are exactly the same.

Example 1:

Lets say x1= 2 and x2 = 5. The difference is 5 – 2 = +3, so the queen should move +3 to reach the pawn. It should stop at the rest square however, so 1 square should be subtracted: +3 – 1 = +2.

Example 2:

If it were the other way around: x1 = 5 and x2 = 2 then the difference would be 2 – 5 = -3, so the queen should move -3 to reach the pawn. This time, it should move one ‘less’ than -3 (that is, -2) to stop at the rest square.

So when the queen moves in the positive direction (ex. 1), one should be substracted from the difference (x2-x1-1) and when it moves in the negative direction, it one should be added to the difference (x2-x1+1). This can be accomplished by subtracting by the quotient of the difference and the absolute difference.

(in formula form): x2-x1 – (x2-x1)/abs(x2-x1) [1]

When the magnet (x3) is already below the queen before starting, x1=x3 and you can also write:

(in formula form) x2-x3 – (x2-x3)/abs(x2-x3) [2]

After that