For the first level, we are asked to check if n coins given by a person are enough to pay for the price of a product. If they are enough, we have to print CHANGE followed by the rest we need to pay back to the customer or if the money paid is not enough, we have to print MISSING followed by the amount of money that's missing.

For the second level, we are asked to check the same thing as in level 1, except this time, we have to print the change in specific coins that are given to us (200,100,50,20,10,5,2,1).

For the third level, the input is pretty similar except now, instead of being given the price for the item, we are being given the entire price grid and the position of the item that the person wants to buy, and we have to select the price from there. Output is the same as in previous levels.

For the fourth level, we have a price matrix and a stock matrix, we will then be given multiple products, and we need to print the total sum of all the products that the person can buy, in order for an object to be purchasable, the object needs to be in stock. After an object is bought, we need to decrease the stock for that product.

For level five, we are given a starting position for a robotic arm and an end position for it. We need to calculate the total time it takes for the robotic arm to move from the starting point to the end point. The robotic arm can move in 8 directions. Because it can also move diagonally, we can easily calculate the total time it takes by knowing it will go diagonally the minimum between the length and the height and then going in a straight direction the rest of the way.

For level six, we have the same question as in the level five, except this time we have a direction blocked. In order to do this level, we have to use dynamic programming and check if we can update the cost to get to any of the possible directions. If we can, we change the cost and add it to the queue again to try and update adjacent directions.