1. What were your results from compare\_cow\_transport\_algorithms? Which  
   algorithm runs faster? Why?  
   2. Does the greedy algorithm return the optimal solution? Why/why not?  
   3. Does the brute force algorithm return the optimal solution? Why/why not?
2. The greedy algorithm finishes the cow transport in 6 trips. The brute force finishes in 5. The greedy algorithm is noticeably faster.
   1. I wrote a small loop that runs and times the algorithms repeatedly and averages the proportion of time taken by the brute force algorithm to that of the greedy algorithm. It averages around 1000 times faster for greedy.
3. No, it only returns the best answer the simple heuristic can find but doesn’t explore all possible answers
4. Yes, it explores all possible answers.

1. Explain why it would be difficult to use a brute force algorithm to solve this problem if there  
were 30 different egg weights. You do not need to implement a brute force algorithm in order to  
answer this.  
2. If you were to implement a greedy algorithm for finding the minimum number of eggs  
needed, what would the objective function be? What would the constraints be? What strategy  
would your greedy algorithm follow to pick which coins to take? You do not need to implement a  
greedy algorithm in order to answer this.  
3. Will a greedy algorithm always return the optimal solution to this problem? Explain why it is  
optimal or give an example of when it will not return the optimal solution. Again, you do not need  
to implement a greedy algorithm in order to answer this

1. A brute force algorithm would need to not just explore, but also store every possible legal result.

2. For a greedy algorithm to the coin problem I would choose the most of the largest value of coin that will fit within the target, then pass the remaining target left and the next largest coin in the sequence into the next recursive call. This can also be done iteratively

3. A greedy algorithm will always provide an optimal solution because the problem is very simple and the simple heuristic of choosing the most of the largest value coin then repeating with each coin will always give the smallest number of coins to produce that value.