# 1 question 1

we are going to prove that this summation $1 + \sum_{k=5n}^{7n} k$ is $O(n^2)$. We will do this by proving that the summation is equal to $12n^2 + 6n + 1$ then prove that $12n^2 + 6n + 1$ is $O(n^2)$

## 1.1 IH

$1 + \sum_{k=5n}^{7n} k = 12n^2 + 6n + 1$

## 1.2 Basecase

n=0
$1 + \sum_{0}^{0} k = 12(0)^2 + 6(0) + 1$
if we simplify the above statement we get
$1 = 1$
so we know that this base case works

## 1.3 IS

we are going to prove that
$1 + \sum_{5(n+1)}^{7(n+1)} k = 12(n+1)^2 + 6(n+1) + 1$
$= (12n^2 + 6n + 1) + 24n + 18$

notice that the n+1 summation is the original summation plus the values 24n + 18. We need to show that
$1 + \sum_{k=5n}^{7n} k + 24n + 18 = 1 + \sum_{5(n+1)}^{7(n+1)} k$
if we write out each summation we will get the following for the n case and the n+1 case respectively
$5n + (5n+1) + (5n+2) + (5n+3) + (5n+4) + (5n+5) + ... + (7n)$
$(5n+5) + ... + 7n + (7n+1) + (7n+2) + (7n+3) + (7n+4) + (7n+5) + (7n+6) + (7n+7)$
here we will notice that all the terms in the middle are the same and that the differences are between the ends of the two summation. If we are to find the differences between the two we just need to subtract the right end of the n+1 case from the left end of the n case. we get the following.
$(7n+1) + (7n+2) + (7n+3) + (7n+4) + (7n+5) + (7n+6) + (7n+7) - [5n + (5n+1) + (5n+2) + (5n+3) + (5n+4)]$
When we simplify this expresion we will get 24n + 18. This means that $1 + \sum_{k=5n}^{7n} k + 24n + 18 = 1 + \sum_{5(n+1)}^{7(n+1)} k$
And this is the thing that we were trying to show. This proves that our assumption was correct. Now we know that the summation is equal to the polynomial $12n^2 + 6n + 1$ we need to show that this polynomical is upperbounded by $n^2$ or that it is $O(n^2)$. to do that we need to find a c and k such that $12n^2 + 6n + 1 \leq cn^2$ for $n > k_0$. if we choose 19 as our c and 1 as our k we can rewrite the inequality as
$12n^2 + 6n + 1 \leq 12n^2 + 6n^2 + n^2$
(we just expanded the $19n^2$ term so that we can clearly show that his inequality holds). This satasfys our definition of something being $O(n^2)$ We have now proven that this summation $1 + \sum_{k=5n}^{7n} k$ is $O(n^2)$

## 2 question 2

assume that this algorithm rearranges the two lists such that there exists an in hospital worker X that has lower priority for a vaccination than an at home worker Y.

notice that in this program every element is compared to every other element. We know this because the for loops will compare an item in the list to every item lower on the list. The rest of the proof considers the sub scenario where an in hospital worker X is compared to a in house worker Y.

There are two senerios

1) X was already higher ranked than Y. we can see that nothing is done in the code when X is already higher priority because non of the if statements will be statsfyied

2) Y initially was higher ranked than X. line 4 says that if an at home worker is higher on the priority list than the at hospital worker(closer to 0 on both of the locations and rank array) their rankings will be swapped so that hospital worker will be higher priority after that if statement body is executed (lines 5, 6).

becuase we know that every item in the list is compared to every other item and that whenever a hospital worker is compared to an at home worker that the hospital worker will end up with higher priority, we know that there cannot be a in home worker that is higher on the priority list than an in hospital worker. This contradicts our assumtion that there exists an at home worker with higher priority than a in hospital worker. This naturally will produce as list with all in hospital workers higher on the priority than the at home workers.

# 3 question 3

to start this algorithmn I will have an initially empty list T that I will update each time I encounter a different type of cow.

For each cow I will compare the type by asking if you are a type that is already in T. If its not already in T then I will add that type to T. By the end of this sequence T will contain all the different types of cows and the length of T will be the variable K discussed in the question. This part of the algorithmn does O(nk) work because we loop over n cows and asking each if its type is contained in T, which T is size K in the worst case.

now we are ready to sort the cows into a container which we will call S. starting at the first element of T, or first type of cow, we will ask each cow if they are this type. if yes is the answer then add it to S.

then we will move onto the next element of T repeating the process. This process will repeat K times and because each iteration of T we are only adding 1 type cow to S we know that each cow of a certain type will be grouped together. This part of the algorithm does O(nk) work because we are looping over the types of cows, which is k work and for each iteration of that loop we are looping over the cows, which is n work.

This algorithm in total is O(nk) because this algorithm has two parts both of which are O(nk), meaning it is 2 * O(nk) which simplifies to O(nk)