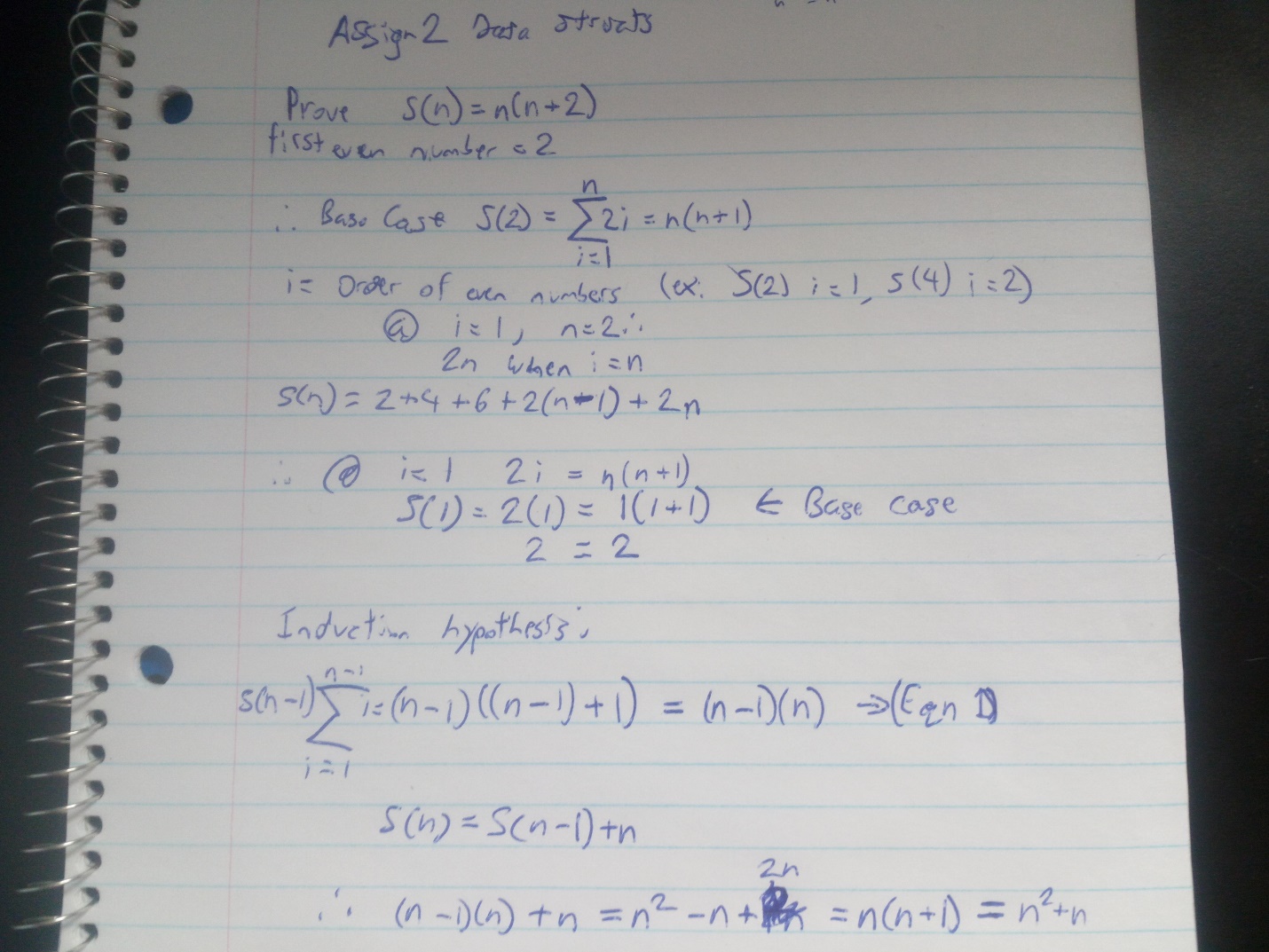
Part 1.

a)

Input array A, int n  
i=0  
sum=0  
for I to n  
 sum->sum+A[i]  
 {increment i}  
return sum

b)



c)

input n even numbers  
x=0  
sum=0  
i=0  
for I to n  
 x=x+2  
 sum +=x(x+1)  
 {increment i}  
return sum

d) The estimated efficiency for the algorithm in part a is O(n) while for part b it’s O(n^2). So the number of primitive operations for each algorithm at n=10, n=100, n=10000 and n=1000000 are as follows:

|  |  |  |
| --- | --- | --- |
| N | O(n) | O(n^2) |
| 10 | 10 | 100 |
| 100 | 100 | 10000 |
| 10000 | 10000 | 100000000 |
| 1000000 | 1000000 | 1000000000000 |

The mathematical induction in b is important as it allows us to represent the same function in a different way, showing that we can have different algorithms for the same purpose and they will have different levels of efficiency.