Section 12.9

1. Logical Approach Comments
   * We both used while loops but the solution used a do-while loop
   * I used a Boolean variable for my while loop condition
   * we both used for loops to iterate through the output
   * A do-while loop is probably more concise, as it required less code, however I do like using a Boolean in the condition for loops as I think it makes the loop condition clearer.
2. Style Comments
   * One thing that differs in our style is white space, I have a lot more of it because I think it makes the code easier to read, also, the solution declared the loop-variable in the condition for the for loop, I declared it under main(). I think what the solution did is better, and I have since started doing it in my code (by since I mean in labs).

Section 12.10

1. Logical Approach Comments
   * The solution used a user-defined function to get no null string
   * We both used two vectors for the time and task vectors
   * The solution used a for loop, I used a while loop to iterate through asking for to do items
   * We both used for loops for the rest of our code
   * For most of the code I think it is mostly sixes, where I think the solution was a lot better was checking for a null string. I didn’t think to do that, and while error checking wasn’t part of the question, if it had been, I’m not sure if I would have thought of doing it that way.
2. Style Comments
   * I actually got docked on this question for my vector names, and look at the solution now, my vector names are definitely lacking, work on being more descriptive.

Section 12.11

1. Logical Approach Comments
   * I used an if-else statement inside my Dominance function. If greed, power = true return true, else return false, the solution just returned the value of green and power. While this is shorter, I overall think my solution is better as to me it is more clear what it returns in which situation.
2. Style Comments
   * Most of this program was pre-written and we have the same style besides that.

Section 12.12

1. Logical Approach Comments
   * The way that the solution and I approached the question was almost the same with one difference, in their multiplysecondhalfinclusive function they just called the multiplysecondhalf function because the conditions ended up being the exact same, looking at my code I used the same formula for both functions, so that was definitely better. Personally, however, if I were to do what the solution did, I would declare a new integer variable, assign it with the value of the multiply second half function, and then return the variable, I think that makes it more clear.
   * I also wrote a bunch of test cases which 1. Helped me understand a lot more how that works, and two helped me figure out if my functions were working correctly.
2. Style Comments
   * Again, I need to declare my loop variables in the loop condition expression. I also think I could have come up with some better names for my two vectors. Besides that, style is about the same.

Section 12.13

1. Logical Approach Comments
   * I included “using namespace std;” which I feel like was a better choice because to me having “std::” in front of every standard library function is messy.
   * In the “vector to string” function we both passed our vector by reference using const
   * The way we approached converting the vector to a string was a little different though, I had if – else statements inside my for loop, whereas the solution had the for loop inside of a if statement. Inside my function I declared another string vector that I iterated through assigning each element of the vector to the string version of the vector we needed to convert. And then I iterated through another for – loop finally assigning a string variable the contents of the string vector. As you can see my solution was long and so the solution was better there (far shorter), however, I think mine was still easier to understand from a reading perspective.
   * Our compute next Fibonacci number was exactly the same except I declared a new integer variable assigned it with the value of the sum of the parameters and returned it. As stated above in 12.12, I think this is a better option. Easier to read.
   * Finally for computing the Fibonacci sequence of a certain size, we had similar approaches, again the solution used if statements before entering the for loop I did not, that meant my loop variable was assigned with 0 in the condition instead of 2, which I think is a better option then using the if-else statements in the loop itself to assign the first and second values of the sequence
2. Style Comments
   * Declaring loop variables in the condition, for testing my functions I used assert again, while the solution tested the output of the sequence. Besides that I think we had similar style

Section 12.14

1. Logical Approach Comments
   * Our code was word for word the same.
2. Style Comments
   * however, I was talking to my dad, (not about the exam, getting help on a lab) he is a senior software engineer and he says he isn’t a fan of using the “this->” in the public member function definitions. Says it is bad style because it is only needed if we use the same name for parameters as we do for the private class members. So maybe one thing I could improve is starting to use different names for the member function parameters.