Symbolic Execution

Problem 2. Symbolic Execution. Consider the code shown here.

```
0.  foo(int x, int y){
1.   if( x > y){
2.     x = x + y;
3.     y = x - y;
4.     x = x - y;
5.   }
6.   if(x > y){
7.     assert(false);
8.   }
9. }
```

Complete the following:

i. For these four function calls, what are the values of `x` and `y` when the concrete execution has exited the if-block at line 5 and is about to execute line 6?

   a. `foo(10,20)`
   b. `foo(42,42)`
   c. `foo(20, 10)`
   d. `foo(16,15)`

ii. Describe in words what this program does.

iii. Let `X` and `Y` be symbolic values representing `x` and `y`. Draw the symbolic execution tree that results from symbolically executing `foo(X,Y)`.

iv. According to your symbolic execution tree, is it possible for there to be an assertion violation? If yes, provide concrete values of `x` and `y` that would cause the error.(Assume that we think of `x` and `y` as infinite precision integers. I.e. there is no integer overflow.)