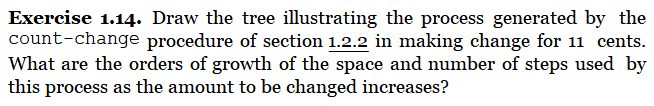
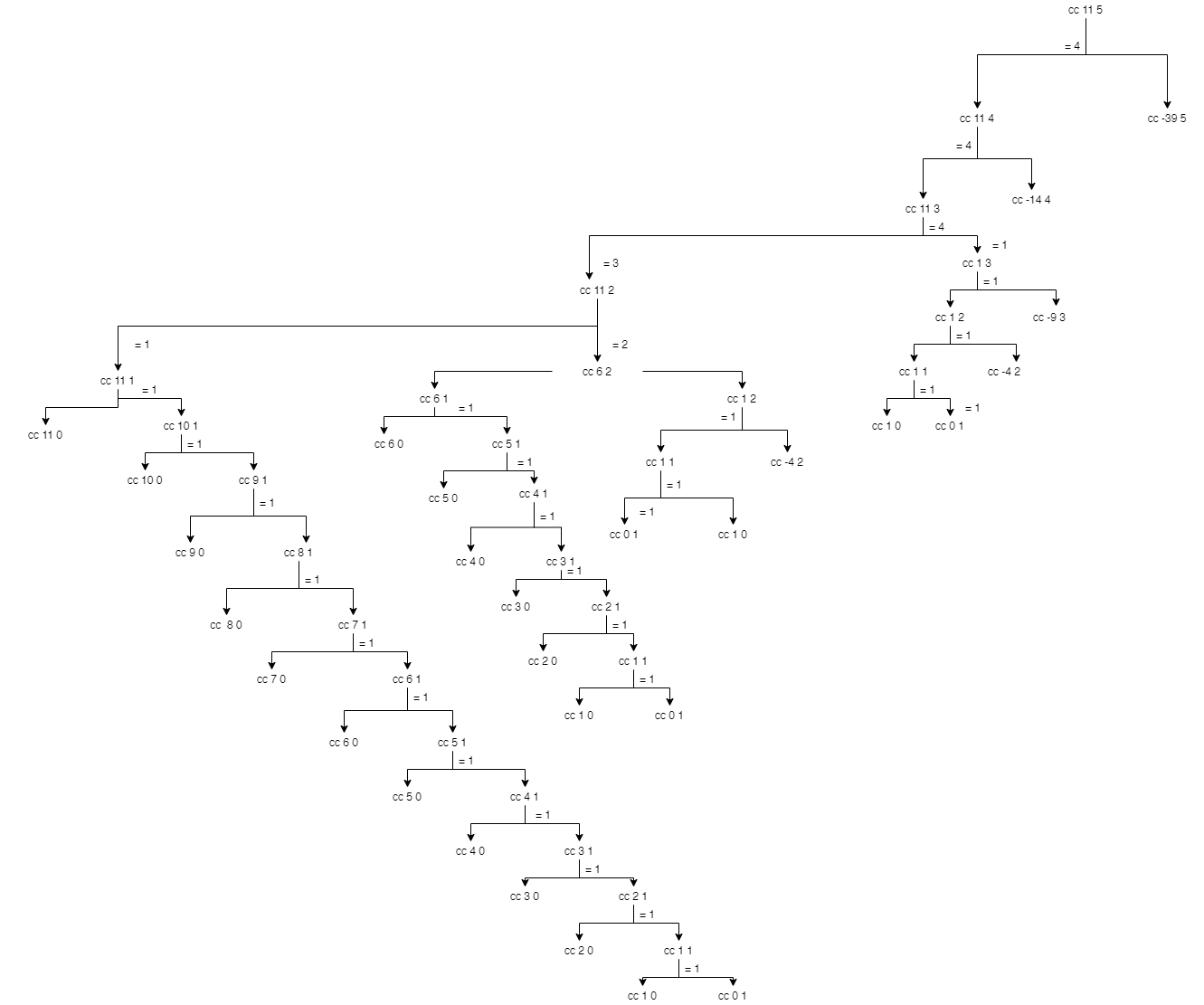
**Structures and Interpretation of Computer Program**

**Exercise Chapter 1.2 Name:** Wan Huzaifah bin Wan Azhar

**Exercise 1.2.3 Orders of Growth**





Time Complexity = O(2^n), each call split into two, if n is large enough, each call will inevitably split into two call, of which both of it will call itself into another two call.

Space Complexity = O(n), at the end of n, which is kinds-of-coins, it will lead to 1, which will expands to amount \* n

1. Procedure P is applied 5 times when (sine 12.15) is evaluated

(sine 12.15)

(p ( sine 4.05))

(p (p (sine 1.35) ) )

(p (p (p (sine 0.45) ) ) )

(p (p (p (p (sine 0.15) ) ) ) )

(p (p (p (p (p (sine 0.05) ) ) ) ) )

(p (p (p (p (p (0.05) ) ) ) ) )

1. Order of growth, space is theta(a) while number of steps is theta(log2 a)

As seen by process generated in a, as a (of angle) gets larger, the procedure p( ) is call more, therefore, the stack will adds up linearly.

The number of steps of n is 1/3 of each call to procedure p. Therefore, the number of steps is reduced one of third each call.