1(a) We begin by assuming P(n) as moving n disk one place counterclockwise, and R(n) as moving n disk two place counterclockwise.

Then if we want to move P(n), which means move n disks one place counterclockwise, we have to move R(n – 1) [move the n – 1 smaller disks two places counterclockwise], move the largest disk one place counterclockwise, then move all the n – 1 smaller disks two places, (back to the top of the largest one), then P(n) = 2R(n – 1) + 1

Similarly, if we want to move all the disks two places counterclockwise, we need to move R(n – 1), then the largest to the second one, then P(n – 1) (move n – 1 disks one place counterclockwise), then the largest to the third one, then R(n – 1) (move n – 1 disks two places counterclockwise). Then R(n) = R(n – 1) + P(n – 1) + R(n – 1) + 2. Since P(n – 1) = 2R(n – 2) + 1, then R(n) = 2R(n – 1) + 2R(n – 2) + 3

We draw a picture to help better understand this:

Our algorithm is:

R(n): Move n disks two peg counterclockwise

P(n): Move n disks one peg counterclockwise

P(n):

Base case: if n = 1

P(n) = 1

Move the disk 1 one peg counterclockwise

General case: if n > 1

P(n) = 2R(n – 1) + 1

R(n):

Base case: if n = 1

R(n) = 2

If n = 2

R(n) = 7

General Case: if n > 2:

R(n) = R(n – 1) + 1 + P(n – 1) + R(n – 1) + 1

= 2R(n – 1) + 2R(n – 2) + 3

Correctness:

Base case: In the picture above, we can see that the base case holds for both R(n) and P(n).

Induction steps:

Assume R(n) and P(n) holds for moving n disks counterclockwise. For n + 1 disks, first we move all the n disks two peg counterclockwise, then n disks to position 2. Then we move the largest to the position 1. Then we move all the n disks one peg counterclockwise, then n disks to positon 0. Then we move the largest to the position 2. Then we move the n disks two peg counterclockwise. Which is R(n – 1) + 1 + P(n – 1) + R(n – 1) + 1. Then the algorithm is correct.

2(b)