```
public int gcd(int x, int y) {
        while(y!=0) {                       →   log (n)
            if (x>=y && x!=0) {             →   1
                int temp = x;               →   1
                x = y;                      →   1
                y = temp % x;               →   1
            }
        }
        return x;                           →   1
}
```

Therefore:

O (gcd(x, y))

= log (n) (1 + 1 + 1 + 1 + 1)

= log (n) (5)

= 5 log (n)

Change all constants to 0

O (gcd(x, y))

= log (n) (1 + 1 + 1 + 1 + 1)

= log (n) (5)

=5 log (n)

Change all coefficients to 1

O (gcd(x, y))

= log (n) (1 + 1 + 1 + 1 + 1)

= log (n) (5)

=5 log (n)

=log (n) → Largest

O (log(n)) = LOGARITHMIC

```
public int hanoi(int n) {
        while (n!=1)                          →   log (n)
        {
            if (n>1)                          →   1
                return 2 * (n-1) + 1;         →   1
        }
        return 1;                             →   1
}
```

Therefore:

O (hanoi(n))

=log (n)(1 + 1 + 1)

=log (n)(3)

=3log (n)

Change all constants to 0

O (hanoi(n))

=log (n)(1 + 1 + 1)

=log (n)(3)

=3log (n)

Change all coefficients to 1

O (hanoi(n))

=log (n)(1 + 1 + 1)

=log (n)(3)

=3log (n)

=log(n) →Largest

O (log(n)) = LOGARITHMIC