**BASE CONCEPTUAL**

**GREEDY ALGORITHMS**

La traducción mas precisa y que nos da más pistas sobre concepto de greedy algorithm es algoritmo codicioso, es llamado así por la manera que este tipo de algoritmos se implementa, es así como dentro del algoritmo hay varias faces y en cada una de ellas se intenta solucionar el problema global de manera local, es decir, en cada uno de los pasos que se da en la ejecución del algoritmo se toma la que parece la mejor solución en un conjunto pequeño de posibles soluciones, es así como podemos hacer una analogía del funcionamiento del algoritmo con una persona codiciosa que intenta encontrar la estrategia con mayor ganancia, este personaje cada vez que vea una oportunidad de ganar algo mas la va a tomar de manera instantánea sin tener en cuenta las consecuencias de la decisión que tomo y de las que dejo de tomar, el ideal de un codicioso es conseguir lo mejor lo más pronto posible.

Podemos ampliar la visión de la situación que estamos planteando, imaginemos un árbol y en cada uno de sus nodos un valor que representa una ganancia para nuestro codicioso personaje y a este lo pondremos en la raíz de nuestro árbol este intentara conseguir la mayor ganancia, así la solución a la que llegaría se vería más o menos así:

3

1

6

4

7

12

2

3

1

6

4

7

12

2

3

1

6

4

7

12

2

En este ejemplo la codicia triunfo y encontró el camino que lo llevo al mayor valor del árbol es decir la mejor ganancia, note que en cada momento se toma el camino que sugiera un valor mayor al actual, 6 > 3, luego 12 > 6, encontrando como máximo valor el 12, pero podemos modificar un poco el problema y el resultado puede ser diferente.

3

1

6

4

15

12

2

3

1

6

4

15

12

2

3

1

6

4

15

12

2

En este caso se llego a la misma conclusión, pero está realmente fue una buena ganancia no la mayor dentro del árbol al igual que en ejemplo anterior la codicia lleva a tomar la mejor decisión dentro de las opciones inmediatas, esto nos da una idea bastante clara del objetivo de implementar un algoritmo de este tipo.

En palabras de Mark Allen Weiss “When the algorithm terminates, we hope that the local optimum is equal to the global optimum. If this is the case, then the algorithm is correct; otherwise, the algorithm has produced a suboptimal solution. If the absolute best answer is not required, then simple greedy algorithms are sometimes used to generate approximate answers.” (P. 499, Data Structure and Algorithm Analysis in C++ , 2014)