**Greedy Algorithms**

It is one of the most intuitive algorithms. We human are greedy so we go for it always.

Greedy approaches are quite simple and easy to understand/formulate.

* Greedy Algorithm always make the choice that looks best at the moment.
* You hope that by choosing local optimum at each step, you will end up at a global optimum.

Suppose you want to have minimum coin change for rupees 39?

So, I will choose a 20 rupees coin then 10 rupees coin then 5 rupees and 2 coins 2 time.

20+10+5+2+2=39

Greedy approach is feasible for Indian currency.

Greedy is faster than DP. But greedy isn’t always correct for all problem.

**HUFFMAN ENCODING**

It is an application used to compress the data and store in the file. And later it is decoded to read it. It is used for space optimization.

We have a string “abbcdda”. Each character takes 2 bytes.

So, 7\*2\*8=112.So it is 112 bits.

So, to optimize space we use Huffman Encoder.

In Huffman Encoder we will create two HashMap’s. One hash-map will be knowns as encoder and another one will be known as decoder used for decoding the data.

**Encoder**

It will have key of type of character. It value will be string. For each character we will have a unique value.

**Decoder**

In decoder side we will have string as a key and character as a value.

**Steps: -**

1. The first step involved in Huffman Encoding is pass the string in the feeder.
2. Make the frequency map of characters.
3. For each key in frequency map create a node and insert in Priority Queue or Min Heap.

char data;

int frequency;

node left;

node right;

1. Loop through the heap. Remove 2 elements from heap and we will combine them until the size of heap is remaining is 1.