Main game of BIT tree is that of the Mange of elements its going to Store. the inventor of Fenwick tree gave a very important unit with which we can play. Range For any index [12] suco [5] suco h sold h BIT [index] is going to store a range of elements from [3] ma [3] mo [3] What is the # of elements its going to Store? Range = [index & (-index)] (Only thing in 614 tile. Bit tree is a manipulation of for any number this is the Least Significant Bit and of LSB. If you get this its Binary representation you understand LSB.

([indask(findus)])

| Storage | Do Barra to | 1.45 21 Dut TIST From Moth |
|-------------|-------------|--|
| index | binary L | SBScm which 13 Stored. |
| | - | |
| traingrain | 000 V | Companier francisco source for more of |
| 2 | 0010 | 2 avr [2] moravi [i] to inter obligation |
| 3 | 0011 | 1 ans [3], arr [2] |
| 9,00 | 0100 | 4 ovr[4], arr[3], ovr[2], ovr[1] |
| 5 Summas. S | 0101 | ang[5] et period is [seloni] TI3 |
| 6 | 0110 | 2 an [6], an [5] |
| 7 | 0111 | The state of the s |
| 65 | 1000 | 8 an[s], an[7], an[6], an[5], an[7] |
| 9 | 1001 | 1 our [9] a |
| 10 | 1010 | $\frac{1}{2}$ $\frac{1}$ |
| 17o-(| 8 . 144.1 | 7110 |
| BIT Indes | Jidinou | ing to Store infra of the nange |
| | 8 7 | s], ans[index-1], ans[index-2], |
| | ~~~ | ([indexk(-index)]) |
| | | (Mange) |

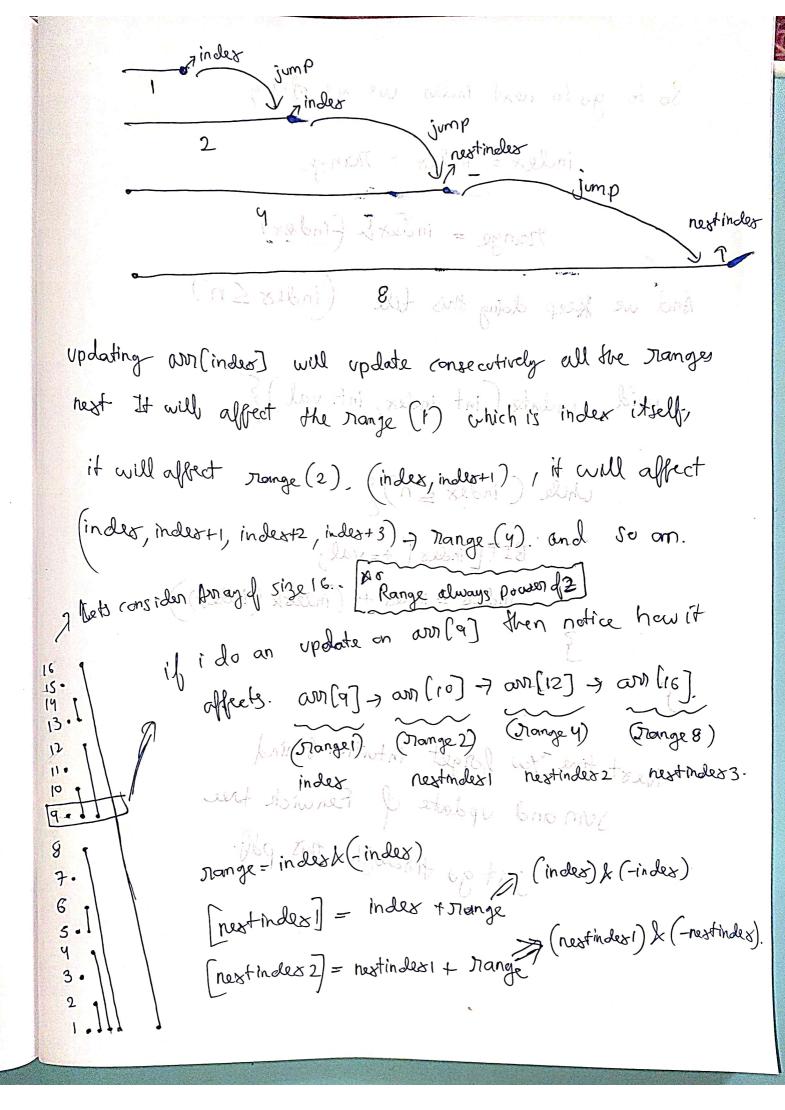
| Letts under Stand Sum Function. Little |
|--|
| As its more intuitive. |
| sum (index) |
| of aggregation (index) => Should give me (p to index) all aggregated details. |
| lets say index = 7 T = 0 111 = BIT [7] = arr[7] Sterled in BIT [7] et (802 [1718] or berry plants of grant states of the state |
| Sout (Contra as panno) Leponto 3) Lucu Emp. In |
| 2 3 9.03 hongs 27 freel, as at but the |
| x 25/1 |
| Now tell me) Will the info of an enr [7] Stored in any hode / index where (index (7) 7 will it be |
| Now tell me -> Will the info of on over [7] Stored in any |
| hode / index where index (7) 7 will it be |
| No Tight. Because BIT [index] Stokes details from |
| aus [index] to aus [index - nange +1] |
| Mentindex = index - (indux ki - index) |
| nange = (index & -index) |

| 1 2 3 4 5 6 7 > |
|--|
| 1234567 |
| 01767 |
| The state of the s |
| BIT [indes] - om [indes],, and indes - nange +1] |
| (range = indest/-indes) |
| (F) Marie Tono is also (F) Marie (F) TO) 6 1110 6 |
| If this mange is already covered in BIT (index), the |
| ar heed to do restly the searching. |
| |
| 1index |
| (index) already covered by (index) index) already covered by (BIT tree (Control of the control of the contro |
| 20 to like if (=) sile of events seleniful |
| So we can move der index to a position where rest of the |
| details are started |

agrice scionifico of --- (xebnifico new Index = index - (index -index) (selone of see in range lot

| Since newIndles = incles - range |
|---|
| ancindes], our (index-range 4) |
| Sowe have to reach on index where ho details of |
| (in dox . Donnas + 1) is there. |
| So what is that index. =) [index-range +1] |
| [albert 1728] of low loop I fi won rebriew of on your previous findly - Trange ore tome of or creations do the loss each TES por gritarific to z work |
| here we more to index -= index & (-index). |
| Chris is nothing but the LSB) |
| Each jumpabackwards we are just reducing the removing the LSB from the binary representation of inde index |
| Firste bit meaning, this will be aways a process of all |
| (index) (index) (index) (index) (index) |
| [index2] [index1] [index] [index] |
| We keep doing it till (index) a) |

| Now that we have a solid understanding of |
|---|
| Now mod or look at update. |
| Sum, Lets look at update. |
| 1 2 6 3: - + - en (index) = -10- 10-051 = -5-1- N |
| servit et (1 + 25 mat - 28 b. i) |
| Lets suppose you are updating (index, tral) |
| we have to wonder now if I add val to BIT[index] |
| Minus |
| how is it affecting my BIT tree and which includes to be exact |
| (s it affecting & xelvi) = - xelvi - xelvi - you need to underetord |
| to understand update first you need to understand |
| Trange |
| Panas = index (-index) = this is always a los |
| Mange = index & (-index) => Misuis always and 32 |
| Single bit meaning, this will be always a power of 2. |
| nanges will be like inther |
| Toestindes 2 |
| (see 1) (see 1) next index) & (see 1) |
| nestandes |
| 8 7 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 |



So to go to next index we are doing index = index + Mange nange = index & (index) And we keep doing this till (index &n) regeries at the glovinosanos etabou the Fredmillion prinche void update (int index, int val) } while (index Zn)} BIT [index] t=val; index = Index + (index (index)) if I do on update on world] then Fotos how it 1. 2) mas & [21] mo & [01) mo & [p) mo & duffo Next time You parget intuition behind Sum and update of Fenwick tree (seli-) & (sepin) Just da Aprograph of this boll. of securities) of (industries) (Second + instantives = [second their