```
Asyun . A. S
14/10/2020
                             Advanced
                                                                 1BM18CS019
                                    Data Structury
wednesday.
                      Week-4 Writing.
                      AVL - Trees.
     11 AVL 19me node
        class Node
                    public:
                         int data;
                        Mode * left;
                        Node & Fright;
                        int height;
       class AVL
           Noch + getnode (ant data)
                    node of = new Node ();
                    P -> data = data;
                    p -> left = NULL;
                     P-7 sight = NULL;
                    p + hught = 1
                     notwor p;
          Node Jow taleleft (Node + a)
                  Node + b = a - right;
                  Node + 6 = a -> lyt;
                     b-> lyt=a)
                     a + sight = s;
                   a > height = max (height (a > suight), height (a + left))+1);
                   b-> height = max (height (b + left), height (b-7 night) +1;
```

getwen b;

Astr St. S

```
Agun A.S
                                                            18 m18 (5019
  Node * sustatusight (Nod * b)
<
        Node *a = b > left)
        Node #t = b - sught;
        a-shight = b;
           b→lyt=t;
       b+ hight = max (height (b > left), height (b > right))+1;
        a > height = max (height (a > left), height (a > right)) +1;
       seturn a;
3
        balance factor (Noch +p)
  int
           if (p == NULL)
                  Setwon 0;
            return height (b > left) - hight (b > right);
        * insection (Node * swot, int data)
           if ( groot == NULL)
                   outurn gebrode (data);
          if (data < 900t -> data)
                  noot > left = invention (noot > left, data)
          Che if ( data > noot -> data)
                  noot -> sught = nusition (noot -> suight, data)
              return scoot
         Swoot -> height = max (height (woot -> left), height (root -> +ight)) +1
        int bataire = balance-factor (root)
       if /fact >1 &4 data < swot > left > data)
                   return notatinight (2000+);
                   < -1 44 data > 2006 -> right -> data)
          (fact
                            notatelft (swot);
                    getwin
```

```
Mayin AS
   if (fact > 1 ss data < 9000t > left > data)
                                                                 IBM 18 CSO19
          return root -> left = notablift (noot -> left)
               Julian notationight (2000)
   if (fact <-1 ff data < 9000t -> left -> data)
               root -> right = notateright (noot -> right).
                return notablift (noot)
    return noot
Node +deletion (Node + swot, int item)
        if (swot == NULL)
               suturn mot.
         if [ item < 900t > data)
                scoot -> left = deletion (scoot -> left, item);
          elu if (item > 900t -> data)
                 noot → sught = deletion (swot -) right, item);
          & if (swot > lift == NULL || swoot > right == NULL)
                 Node + & = root - sight ? snoot -> sight : snoot -> left ;
                    if ( == NULL)
                           s=9000t
                           root = NULL
                    elu + root = + temp)
                     free (temp);
              else
                  Node At = minvalue (900t → siight)
                  noot >data = t → dala
                  swot > sight = deletion (swot > right, t > data)
```

-3 -

```
int fact = balance factor (9000t)

balance factor (9000t)

if (fact 71 & & Quee (9000t > left) 7 = 0)
                                                                 Arjun. A. S
          reliver rotateright (400t);
if (fact <-1 84. balance-factor (scoot -> left) <=0)
           return notatelet (noot);
if ( fact >1 & & balance factor ( root > lift) < 0)
             not > left = swetateleft (900t > left)
              guturn sustatisight (1006);
 if (fact <-1 44 balance-factor (900t > 90ght) 70)
                 noot > night = notatinight ( noot > night);
return notatilyt (noot);
  neturn noot;
```

1BM18CSD19