AVL TREE

```
NODE * rotateright(NODE *x)
{
  NODE *y;
  y=x->left;
  x->left=y->right;
  y->right=x;
  x->ht=height(x);
  y->ht=height(y);
  return y;
}


NODE * rotateleft(NODE *x)
{
  NODE *y;
  y=x->right;
  x->right=y->left;
  y->left=x;
  x->ht=height(x);
  y->ht=height(y);
  return y;
}


 NODE* insert(NODE *T, int x)
 {
   if(T==NULL)
   {
     T=(NODE*)malloc(sizeof(NODE));
     T->data=x;
     T->left=T->right=NULL;
   }
   else
   if(x > T->data)
   {
     T->right=insert(T->right,x);
     if(BF(T)==-2)
     if(x>T->right->data)
     T=RR(T);
     else
     T=RL(T);
   }
   else
   if(x<T->data)
   {
     T->left=insert(T->left, x);
     if(BF(T)==2)
     if(x < T->left->data)
     T=LL(T);
     else
     T=LR(T);
   }
   T->ht=height(T);
 return(T);
 }
```