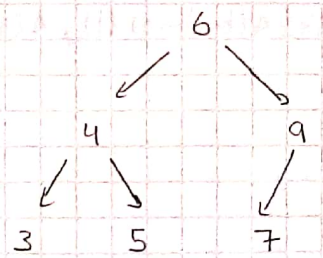
inordent (Nodo<T> * head)	if(head == NULL)	return ;	inorden (head->getIzq())
1) inorden(head(6))	head == Null F		inorden (4)
2) inorden (head(4))	head == Null F		inorden (3)
3) inorden (head(3))	head == Null F		inorden (Null)
4) inorden (Null)	Null == Null V	return 3)	
5) inorden (head(5))	head == Null F		inorden (Null)
6) inorden (Null)	Null == Null V	return 5)	
7) inorden (head(9))	head == Null F		inorden (7)
8) inorden (head(7))	head == Null F		inorden (Null)
9) inorden (Null)	Null == Null V	return 8)	
10) inorden (Null)	Null == Null V	return 9)	
11) inorden (Null)	Null == Null V	return 7)	

cout<<head->getKey()	inorden (head->getDer())
1) cout<< 6	inorden (9)
2)	
3) cout<< 3	inorden (5)
4) cout<<4	
5) cout<<5	
6)	
7) cout<<9	inorden (Null)
8) cout<<7	inorden (Null)
9)	
10)	
11)	



Impresión : 3 4 5 6 7 9

insert(Nodo<T> *head, T x) | if(head = NULL) | temp = Nodo<T>(x) | return temp | if(x < head->getKey())

1)	insert (Null, 5)	Null = Null V	temp = Nodo<T>(5)	return temp(5)		
2)	insert (head, 7)	head = Null F			7 < 5 F	
R2 3)	insert (Null, 7)	Null = Null V	temp = Nodo<T>(7)	return temp(7)		
4)	insert (head, 4)	head = Null F			4 < 5 V	
R4 5)	insert (Null, 4)	Null = Null V	temp = Nodo<T>(4)	return temp(4)		

head->setI(insert(head->gI(), x)) | else if(x > head->gK()) | head->sD(insert(head->gD(), x))

1)
2) 7 > 5 V | head->sD(insert(Null, 7)) => head->sD(temp(7))
3)
4) head->setI(insert(head->gI(), 4) => head->setI(temp(4))
5)

head->sA(1 + max(A(head->gI()), A(head->gD()))) | bal = A(head->gI) - A(head->gD) | if(bal > 1) | if(x < head->gI->gD)

1)
2)
3)
4)
5)

return RD(head) | else head->sI(RI(head->gI)) | return RD(head) | elseif(bal < -1) | if(x > head->gD) | return RI(head)

1)
2)
3)
4)
5)