- 0	4-PD3
	cicio 1:
	guegie Natural - El nelodo debe de comenzar
	vor et Noob que se quior agregor, evalvarb de
_ (rue no este agregado ya en el osbol.
	vego, valdo el arbol y compera si
	el valor del mode que se quiere agragar i
	es mayor o menor que la raiz. si es
	menor comperor con el hijo izquierdo y
	vuelve a prequeter le visno de ontes,
	hesta au lleau a con all al l'in ma
	o perceho, dependiendo de dorde se quieros agregos
	Hace le nison on al anne se queros eigregos
	mayor.
	A STATE OF THE PARTY OF THE PAR
	Precondiciones: by like one el mode no existe
	enter va.
	verifier are d'altrol sea distante de nula
	NV/O
ç	estandiciones: El estal delle mantererse en aquilibrio
	iras la inserció del nodo.
1-	El asbol lebe incrementes one tourne (conf de noute
	Pseudocadigo: Inicio Nodo Arlos
	cyters data
	Cadena, nowber
	Nodo Arbol hijo Fzquiordo, hijo Derecho.
	The state of the s
	Constructor (d, nom)
	dato = d
	powers = non 1
1	hijo Izqu'ordo = nulo

Initio Agricultud Arbel Birmo (d, nom) ratz & rolle contero id string nom linene = crear underted (d, nom) si (mi = = note) horee ra i = nume line linene = verification availar = verification linene = number verification ratio = number verification ratio = number verification ratio = number number number si (d < number = number si (number = number devolver final timal timal Timali		
string nam string	Tak	in Agregor Nodo Albel Bingro (d. non)
string many Elvente = Crear Understand (d. nom) 2: (m/2 = mole) bacar ra i = nuevo Fina Michael Actand a valler = 1012 Microtras (verbodoro) Final (d. x sumiliar doub) bacar 2 unities = number = 2 uniter by 12 galerdo 2 i (sumiliar = number = 2 uniter by 12 galerdo 2 i (sumiliar = number = 2 uniter by 12 galerdo 5 i (sumiliar = number devolver Final		raiz & noto
Provide = crear sude Arber (d., nom) 2: (raiz = 2 mole) horses 1: (raiz = 2 mole) horses 2: (raiz = 2 mole) horses 4: (raiz = 2 mole) horses 4: (raiz = 2 mole) 2: (raiz = 2 mole) 2: (raiz = 2 mole) 2: (raiz = 2 mole) 4: (raiz = 2 mole) 5: (raiz = 2 mole) 5: (raiz = 2 mole) 4: (raiz = 2 mole) 5: (raiz = 2 mole) 5: (raiz = 2 mole) 5: (raiz = 2 mole) 6: (raiz = 2 mole) 6: (raiz = 2 mole) 6: (raiz = 2 mole) 7: (raiz = 2 mole		entero d
Provide = crear sude Arber (d., nom) 2: (raiz = 2 mole) horses 1: (raiz = 2 mole) horses 2: (raiz = 2 mole) horses 4: (raiz = 2 mole) horses 4: (raiz = 2 mole) 2: (raiz = 2 mole) 2: (raiz = 2 mole) 2: (raiz = 2 mole) 4: (raiz = 2 mole) 5: (raiz = 2 mole) 5: (raiz = 2 mole) 4: (raiz = 2 mole) 5: (raiz = 2 mole) 5: (raiz = 2 mole) 5: (raiz = 2 mole) 6: (raiz = 2 mole) 6: (raiz = 2 mole) 6: (raiz = 2 mole) 7: (raiz = 2 mole		strine non
Tingi Ti		MUENO = creas undo Bricol (d nom)
Microtras (verdadera) Fedre = auriliar = vais Microtras (verdadera) Fedre = auriliar (d < auriliar = auriliar, byo = zayderab si (auriliar == nulla) Fadre, hyo = zayderab devolver Finsi sino auxiliar = auriliar, hyoderaba si (auxiliar == null) pedre, hyoderaba devolver Finsi tinsi tinsi Findir Finsi Tinsi		si (miz se mila) hace
Moderated padre Micritian (verdadera) Findre = aurilian (i) (i) (i) verdadera) Findre = aurilian (i) (i) (i) verdadera, hijo transcribe Si (aurilian = aurilian, hijo transcribe devolven Findre = aurilian, hijo tenada Si (aurilian = aurilian, hijo tenada Tinai		
Modelical Padre Miertras (verdadora) Fedre = averius (I d < averiliar = averius hour averiliar =		·
Modelical Padre Miertras (verdadora) Fedre = averius (I d < averiliar = averius hour averiliar =		Note Achal a valier = sais
Migratus (verdodero) Fedre = avvivar (d < avvilier = avvivar avvilier = avviler, hijo regulerdo si (avvilier == nulo) Padre, hijo regulerdo devolver finsi sino avvilier = avviler, hijodere cho si (avvilier = avvil) padre, hijo reccho = nuevo devolver Finsi Tinsi		
Finds		
(d < supplier, doub) haver supplier = supplier, high regulards si (supplier = nulls) Podre, high regulards = devolver fins: sino auxilier = supplier, highere sho si (supplier = null) pudre, higherecho = nuevo devolver Fins: tins: Tins	-	
Si (auxilier = 2 mule) Fadre, hijotresho devolver Finsi Sino auxilier = auxilier, hijoderesho Si (auxilier = auxilier = auxil) pedre, hijoteresho = nuevo devolver Finsi Tinsi Tinsi	-	
Si (auxilier == null) Padre, hijotredo- devolver Finsi Sino Si (auxilier = auxilier, hijoderecho Si (auxilier == null) padre, hijoterecho == nuevo devolver Tinsi	_	auxiliar = auxiliar, his I radierdo
Finsi Tinsi Ti		Si (auxiliar == nula)
Fins: Sino auxilier = queiler hijodere cho Si (quevilier = = null) pedre hijodere cho = nuevo devolver Fins: Fins: Fins: Fins: Fins: Fins: Tipp:mir Arbol		
sino auxilier = auxilier = = autil pedre hijoserecho = nuevo devoluer Tino: Tino:		
auxilier = auxilier = = nu(1) Si (auxilier = = nu(1) padre hijoterecho = nuevo devolver Finsi Finsi Finsi Finsi Tinsi T		fins:
Si (applier = = null) padre hijoperecho = nuevo devolver Tinsi Tinsi		
pedre hijoperecho = nuevo devolver Tinsi Tinsi Tinsi Tinsi Tinsi Tinsi Tinsi Tinsi Topomir Arbol		auxilier = qualter. hijodere sho
Tins: Ti	منسيب	
Tinsi Tinsi Tinsi Findir Findir Tinsi Tinsi Tinsi Tinsi Tinsi Thermir Arbol	18	padre hijo Derecho = nueve
Tingi Finding Finding Finding Finding Tinging Arbol		
Fin Aspentas Fin Si Imprimir Arbol		
Finalize Finalize Tin Si Imprimir Arbol		
Findir Fin Si Imprimir Arbol		
Fin Si Imprimir Arbel		
Imprimir Arbel		
		그는 그 그들은 사람들은 사람들이 되는 것을 가는 것이 되었다면 하는 것이 되었다면 하는 것이 되었다면 하는 것이 되었다면 하는데 그렇게 되었다면 그렇게 되었다면 하는데 그렇게 되었다면 하는데 그렇게 되었다면 하는데 그렇게 되었다면 그렇게 그렇게 되었다면 그렇게 되었다면 그렇게 되었다면 그렇게 되었다면 그렇게 되었다면 그렇게 그렇게 되었다면 그렇게

Orden del Transo de Kjewich del Algoritmo Al ser in expol binerio, el orden del tienpo de ejeanion del Adgardoso, es de allas Al) si esta balancoods, si no esta boloncoolo este es ocuj. Ejertico 2 Contor todas la hojas que tiene el Arbol. Lorgueye Noctoral: Lo que debeno hace o un programa que vaya varificando si cada visola tieve hijos, en la izquierda, en la derecha o en ambas. Si tiene hijos no se la considera hoja si no tiene hijos, se la considera choja por la que aumenta el contedor en 1 postiendo dosde O. Precontiones: [] asbol no este vacio. Tener un contailor para contai les hojos Postcochicions: Devolver un contador. Pseudocodigo: Inicio Confor Hojas (contador i) $i \leftarrow 0$ Hodeochup! si Nedoactual es nulo devotor o imprimir "verificado nodo con vabri", noch. valor tienen Tegularda = "si 4 si nodo. Tegulardo no es nulos sino is time draho : "sittal nabidecho no es nulo sino "no" Imprimir "Hijo Expirate", time Exquedo "Hypo derechoi", tieve durcho

	5
	•
1 mode, 2	Zgdordo)
hojers _ Tzque do = conter hojers y colificas hijos (nodo, z	cho)
11 derection = confer-hojes, verificer-highes (node dere	eno) C
	- 6
es-hopia - 1 si nodo. Izquerdo es mulo y nodo dereal	10
es nulo sino o	
ES NOTO SIND	
	4
devoler sically as hojas - 7 equiado: + hojas - dereha, + 10-, hoja	
Imprimir Contingos	
As A	•
orden de Ejeweión de Algoritmo	6
Tro orden de Ejecución de este Algoritro, es en	,
O(n), en donde n es la contidad de noobs de	•
arbol.	
	4
	•
	(
	•
الأرب حيراب منعت	
the state of the s	
	1
	-