No.	
Date	
(18) Pelcte Atter	
Velcte AFFE	-
Procedure Del-Atter (* plet: address, "x: Intetope).	
Procedure Del-After (* pBex: address, *x: Infetype). If pBet is Null or Next(pBex) is Null Then	
Else Na Cara	
Dedare temp as address & Next (pBef) x & Inxo Ctemp)	
Next (pBet) & Next temp)	_
Realcocase temp	
tud le	0
End Procedure	
	0
	•
	-0
	-
	- 9
	. 0

17). Relete Aktor Procedure Del-Aktor (*P: address: * x:infotype): IP P is NULL Then Else | K Next(p) is NULL Then x + (npo (p) Deallo case P P& NULL El Se Declare prev as address & NULL Declare corn as address or P Whole Nex (carr) is not NULL DO prev a curr carr + Next(carr) End orbite x & lupe (cur) Next (prev) & NUCL Reallocak con Frod l# End Procedure

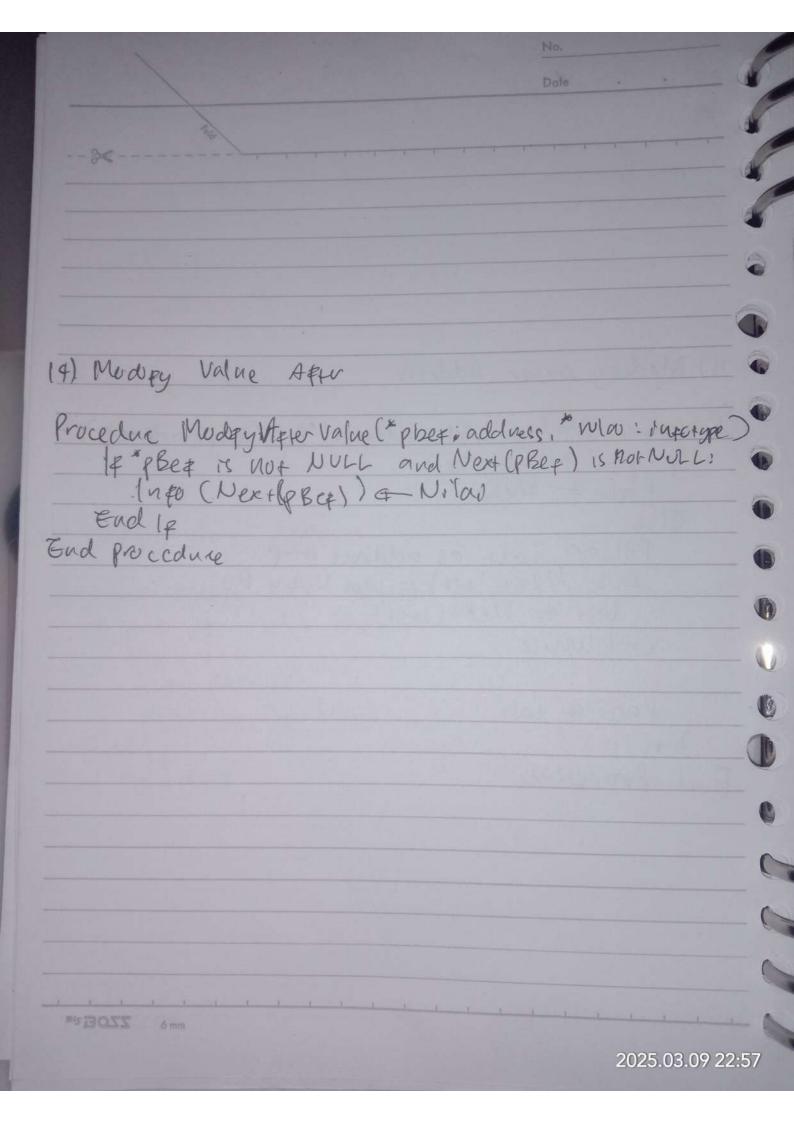
MA ZZOEZ

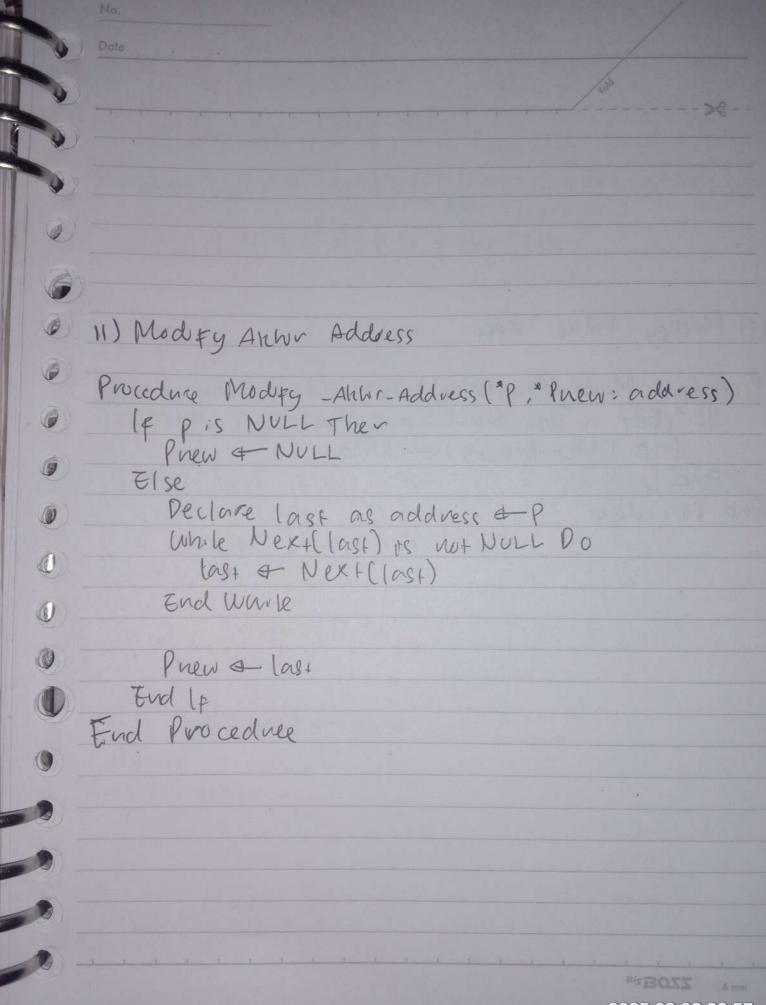
100

2025.03.09 22:58

16) Delete Awal Procedure Del-Aual (* p:address, * * x:intotype): IF p is NULL Then Else Declare temp as address & p x & luto (temp)
P & Next (temp) Reallocate temp End lt End procedure 0 C 0 1 2025.03.09 22:58

15) Modepy Bet Value Betwee Procedure MedifyBerforeValue (* p: address, pAF+: address, enlar: infotype) IF pis NULL or pis patt Then End 18 Declare prev as address & NULL Declare curr as address & P While corr is not NULL and Next (corr) 1sut paper: Preva curv 10 carr or Next (curr) 1 End whole 0 le carr is NULL Then (1) Wa 4-1 Else Into (Prev) + wear End if End Procedure





W) Mudey Awal Address Procedure Medity-And-Address (* P: address, * Pren: ada) Prew 4-P 4 P & Next(P) Next (Prea) - NULL End 14 End procedure

2025.03.09 22:57

8) Insert Value l'axter

Procedure Insert After Value (PBEF: address, whis imports pe)

Else

Reclare Pnew as address & crease Node (w/a) Next (Pnew) & Next (p Bet) Next (pBet) & Pnew

End If

End procedure

miz BOZZ Q wu

5) tous (neers akhir Protedure Ins. Ahim (*p: address, Pnew: address). 1 p is NULL Then &p & Pnew Next (Pren) + NULL Declare last as address & P While Mext (last) is not NULL: last of Mext (last) 10 End whole Mext (last) & Pnew Find 14 End Proedure sig BOZZ 6 mm

4). Insert Awal Procedure Ins-Awai (*p: address, Pnew: address): le p is Mull Then
p & Pnew Else Mext (Pnew) ap Pd Pnew End 14 End Procedure 2025.03.09 22:56

recedence (so Node (p) address, who interpol): The p is NULL Then: Charle Node (p) Into (p) of Mode (p) Mext (p) of Mull Else Declare of as address Charle (p) of Mull Else Persone Node (q) Node (q) Node (q) Postedine	A-Node (*P: address. x: inforge): x Thode (*P: address. what: inforge): Node (P) Out Then: Node (P)	n-Node (*p: address. x: inforge): x Thode (*p: address, what: inforge): Node (*p) A as address Node (a) A p A p Node (a)	m-Node (*p: address. x: inforge): * * * * * * * * * * * * *	m-Node (*p: address. x: infarge): * ** ** ** ** ** ** ** ** *	M-Node (*p: address. x: infarge): ** ** ** ** ** ** ** ** **	Procedure (r. Node (r. address. x. interspe): [upo(p) 4 x End procedure Procedure merry Node (p. address. value interspe): [tp p is NULL Then: [crane Node (p) Mext(p) 4 Node (p) Mext(p) 4 Node (p) Next(q) 4 Node (p) Para (p) 4 Node (p) Find procedure Find procedure Find procedure The procedure Para (p) 4 Node (p) Para (p) 5 Node (p) Para (p) 6 Node (p) Para (p) 6 Node (p) Para (p) 7 Node (p) Para (p) 8 Node (p) Para (p) 8 Node (p) Para (p) 8 Node (p) Para (p) 9 Node
m-Node (*p: address: who is inforge): * ** ** ** ** ** ** ** ** *	ry Node (*p: address. x: inforge): x ry Node (*p: address. mlast: inforge): - Node (p)) a nlas p) a st address Node (q)) 4 nlas	m-Node (*p: address. x: infarge): rt Node (*p: address. what: infarge): Node (p)) of miles Node (p) Node (p) Node (a)	m-Node (*p: address. x: inforge): ** ** ** ** ** ** ** ** **	m-Node (*p: address. x: inforge): ** ** ** ** ** ** ** ** **	ry Node (*P: address. x: inforge): x VULL Them: Node (*P: address. who: inforge):	ry Node (*P: address. x: inforge): x Thode (*P: address. what: inforge): Node (P) Out Then: Node (P) Node (A) Node (A) Node (A) Node (A)
m-Node (*P: address. w. inforgre): Thode (*P: address. w.lav.: inforgre): Node (*P) N	m-Node (*p: address. x: inforge): * * * * * * * * * * * * *	m-Node (*p: address. x: inforge): x VULL THEM: Node (p)) or mini Node (p) Node (q) Node (q) Node (q)	m-Node (*P: address. x: inforge): x r+Node (*P: address. wlav: inforge): -Node (P) -Node (P) -Node (P) Node (A) Node (A) Node (A)	m-Node (*p: address. x: infarge): rt Node (*p: address. what: infarge): Node (p) Out Then: Node (p) A saddress Node (a) Node (a)	m-Node (*p: address. x: infarge): * * * * * * * * * * * * *	m-Node (*p: address. wini inforge): * Node (*p: address. wini: inforge): Node (*p) And NULL Node (*p) Node (a)
n-Node (*P: address. who inforge): x VULL Then: -Node (*P: address. who: inforge): Node (P) Node (P) Node (P) Node (P)	n-Node (*P: address. w. inforge): x Thode (*P: address. w. inforge): Node (*P) P) 4- Noll Node (a) Node (a)	m-Node (*P: address. what inforge): Thode (*P: address. what inforge): Node (*P) Node	n-Node (*P: address. x: inforge): * * * * * * * * * * * * *	m-Node (*p: address: who inforge): * ** ** ** ** ** ** ** ** *	m-Node (*p: address. x: infarge): rt Node (*p: address. what: infarge): Node (p)) of nilar p) of Null Node (a) Node (a)	m-Node (*p: address. x: infarge): ** ** ** ** ** ** ** ** **
Ande (P: address. x: inforge): A Node (P: address. what inforge): Node (P) Out Then: Node (P) A as address Node (A) A Node (A)	m-Node (*P: address. x: inforgre): Thode (*P: address. what: inforgre): Node (P) OULL THEN: Node (P) Outh Then: Node (P) Node (P) Node (P) Node (P) Node (P)	m-Node (*p: address. x: inforgre): * * * * * * * * * * * * *	M-Node (*P: address. what inforge): * VULL Then: Node (P) O - mini P) o - NULL Node (A) Node (A)	m-Node (*p: address. what inforge): * Node (p) O - what Node (a) Node (a) Node (a) Node (a)	Made (*P: address: what inforge): ** ** ** ** ** ** ** ** **	m-Node (*p: address. x: inforge): * Node (*p: address. vulni: inforge): Node (p)) of nilar P) of Noll Node (a) Node (a)
M-Node (*P: address. what inforge): * * * * * * * * * * * * *	m-Node (*p: address. what: inforge): * * * * * * * * * * * * *	n-Node (*P: address. wlav: inforge): Node (*P: address. wlav: inforge): Node (*P) Nod	M-Node (*P: address. who inforge): * * * * * * * * * * * * *	n-Node (*p: address. what inforge): * * * Node (*p: address. what inforge): Node (p)) - Node (p) Node (q) Node (q)) + Note (q)	m-Node (*p: address: who is inforge): The Node (*p: address: who is inforge): Node (p) Out Then: Node (p) And address Node (q) Node (q)	ry Node (*P. address. what inforge): x Y OULT Then: - Node (P)) a nilar P) a NULL Node (A) Node (A) 14 P
m-Node (*p: address. who interpol): * Thode (*p: address. who interpol): Node (p) Out Then: Node (p)	m-Node (*p: address. who inforge): * * * * * * * * * * * * *	Made (*p. address. what inforge): * Node (*p. address. what inforge): Node (p)) a- what Node (q) Node (q) 14- p	Made (*P. address. what inforge): ** Node (*P. address. what inforge): Node (P) P) of what Node (A) Node (A) Node (A)	m-Node (*p: address. who: inforge): * * * * * * * * * * * * *	Made (*p. address. what inforge): ** Node (*p. address. what inforge): Node (p)) a- what Node (q)) a- what Node (q)) a- what Node (q)	Mr. Node (*P. address. wlast inforge): ** ** Node (*P): address. wlast inforge): - Node (P) ** ** ** ** ** ** ** ** **
m-Node (*p: address. who imparge): The Node (*p: address. who is inforge): Node (p) The Node (p)	n-Node (*P: address. wlav: inforge): * Thode (*P: address. wlav: inforge): - Node (P) - Node (P) - Node (P) - Node (A) - Node (A) - Node (A)	M-Node (*P: address. wlas: inforge): * * * * * * * * * * * * *	Man Node (*P: address. what inforge): ** ** ** Node (*P: address. what inforge): Node (P) ** ** ** ** Node (P) ** ** ** ** ** ** ** ** **	n-Node (*p: address. what inforge): ** ** ** Node (*p: address. what inforge): Node (p) ** Node (p) ** Node (p) ** ** ** ** ** ** ** ** **	Made (*P: address. who inforge): * * * * * * * * * * * * *	n-Node (*p: address. what inforge): ** ** ** Node (*p: address. what inforge): Node (p) ** ** ** ** Node (p) ** Node (p) ** ** ** ** ** ** ** ** **
A-Node (*p: address. x: inforge): * * * * * * * * * * * * *	n-Node (*p: address. x: inforge): * * * * * * * * * * * * *	m-Node (*p: address. who is inforge): * * * * * * * * * * * * *	m-Node (*p: address. x: inforge): * * * * * * * * * * * * *	m-Node (*p: address. who inforge): * * * * * * * * * * * * *	m-Node (*p: address. what: inforge): * * * Node (*p: address. what: inforge): Node (p)) a- what Node (p) Node (q)) a- what Node (q)	m-Node (*p: address. who inforge): * * * * * * * * * * * * *
Procedure (&-Node (*p: address. x: inforge): [rolling (p) + x End procedure Procedure meer Node (*p: address. vilor: inforge): [f p is Null Then: Create Node (p) Info(p) 0- min Mext(p) 0- min Else Peclore of as address Create Node (q) [upo (g) 4- vilor Nati(g) 4- p Fod if End procedure	Procedure (n-Node (p; address, x; inforge): End procedure Procedure mert Node (p; address, what inforge): If p is NULL Them: Create Node (p) Info(p) a- who Mext(p) a- who Else Declare a address Create Node (q) Info(p) a- who Next(q) a- p Fud if End piecedure	Procedure (n-Node (p; address, x: inforge): End procedure Procedure mert Node (p; address, what inforge): If P is NULL Truen Crease Node (p) Info(p) of nilar Mext(p) of NULL Else Declare of as address Crease Node (q) Info(p) of vilar Next(q) of p Fud if End procedure	Procedure (n. Node (*p. address. x. inforge). End procedure Procedure mert Node (*p. address. what inforge). If p is NULL Then: Crane Node (p) Info(p) of nilar Mext(p) of NULL Else Crane Node (q) Info(p) of NULL Else Crane Node (q) Notight p Find procedure	Procedure (m. Node (*p. address. x. inforge). End procedure Procedure mert Node (*p. address. ulai inforge). If p is NULL Them: Create Node (p) Info(p) of nim Mext(p) of Noll Else Declare As address Create Node (q) Info(q) of via Node (q) Find procedure	Procedure (n-Node (p; address, x; inforge): End procedure Procedure mert Node (p; address, what inforge): Le p is NULL Them: Crease Node (p) Ango(p) of nilar Mext(p) of NULL Else Declare a as address Crease Node(a) Inpo(p) of vilar Next(q) of p Find procedure	Procedure In-Node (*p: address. x: inforge): End procedure Procedure mert Node (*p: address. what inforge): If p is NULL Then: Crane Node (p) Anso(p) of miles Mexist(p) of NULL Else Crane Node (q) Inpo(q) of what Notig) of vila Notig) of p
Procedure (by Node (p) address, who inforge): End procedure Procedure insert Node (p) address, who inforge): Let p is NULL Them: Create Node (p) Info (p) of nilar Mexit (p) of NULL Else Create Node (q) Info (p) of vilar Next (q) of vilar Next (q) of p	Procedure (by Node (p) address, who inforge): End procedure Procedure merr Node (p) address, who is inforge): If P is NULL Then: Create Node (p) Ango (p) a- Node (p) Next(p) a- Node (a) Next(q) a- Node (a) Find procedure Find procedure	Procedure (m. Node (p: address. x: inforge): [man procedure Procedure merry Node (p: address. mlos: inforge): [f p is NULL Them: Create Node (p) Ango (p) of mlos Mext (p) of Mode (q) [hop (q) of Node (q) Nat (q) of procedure Find procedure [man	Procedure (sr_Node (p; address, who is inforge): End procedure Procedure insert Node (p; address, who is inforge): If p is NULL Trun Crane_Node (p) Anfo(p) of nilon Next(p) of NULL Else Quelow a as address Crane_Node(a) Info (p) of vila Next(a) of p Find procedure	Procedure (n-Node (p: address, what interpe). End procedure Procedure mert Node (p: address, what interpe). If p is NULL Then: Create Node (p) Anto (p) of mini Else Declare 9 as address Create Node (q) Inpo (q) of what Next (q) of p P of q Find procedure	Procedure (m_Node (*p: address. x: inforge): End procedure Procedure mert Node (*p: address. mini: inforge): If p is NULL fruit Creak - Node (p) Info(p) - mini Mext(p) - mini Else Peclore of as address Creak - Node (q) Info(p) + mini End procedure For of Fo	Procedure (m_Node (*p: address. x: inforge): [upo (p) 4-x End procedure Procedure neart Node (*p: address, what inforge): [f p is NULL Then: [crane Node (p) Aufo (p) 4- NULL Else Pectore q as address Crease Node (q) [upo (p) 4- who Next (q) 4- p Find procedure Find procedure
Procedure (s. Node (p; address, x: wearge): End procedure Procedure meer Node (p; address, what: inforge): Let p is NULL Then: Create Node (p) Lufo (p) of miles Mext(p) of NULL Else Declare q as address Create Node (q) Next(q) of powedure P of q End procedure	Procedure (sr-Node (P; address, x: weaspe): End procedure Procedure merry Node (P; address, what: inforspe): If p is NULL Then: Create Node (P) Info(P) of nilar Mext(P) of NULL Else Peclare of as address Create Node (a) Info(P) of Null Find procedure Oracle (A) of P Find procedure	Procedure (sr. Node (p: address, x: measge): End procedure Procedure merry Node (p: address, what: inforspe): If p is NULL Them: Create Node (p) Augo (p) of milan Mext(p) of NULL Else Declave of as address Create Node (q) Into (p) of vilan Not (q) Por of Find procedure	Procedure (sr. Node (P: address, x: weage): End procedure Procedure merry Node (P: address, what: inforge): If P is NULL Then Create Node (P) Ango (P) of miles Mext(P) of Null Else Pectore of as address Create Node (a) Into (P) of vilas Nation of P Find procedure Into (P) of P Find procedure	End procedure Procedure Mercy Node (*P. address. x: measge): End procedure Procedure Mercy Node (*P. address. mlas: inforge): If P is NULL Thum Crane Node (P) Mercy (P) of mlas Mext(P) of NULL Else Crane Node (A) Next(A) of P Page (P) of mlas Next(A) of P Find procedure	End procedure Frowarie ment Node (*p. address. mini inforge): If p is NULL Thun: Crane Node (p) Mexico q as address Crane Node (q) Mexico q as address Crane Node (q) Mexico p + vila Nexico p Fra q Fra procedure	End procedure Procedure insert Node (*p. address, what interpe): If p is NULL Then: Creat Node (p) Anso(p) of man Mext(p) of NULL Else Pectore of as address Create Node(q) Inspo(q) of vilan Noxi(q) of vilan Noxi(q) of p Find procedure
End procedure Procedure mert Node (*p.: address. what: inforge). If p is NULL Them Create Node (p) Info (p) of nilos Mext(p) of Null Else Create Node (a) Info (g) of vilas Next(q) of p P of g Find precedure	End procedure Procedure merit Node (*p.: address. what: inforge). If p is NULL Thum Crane Node (p) Info(p) a- nilar Next(p) a- nilar Next(p) a- Node (q) End procedure End procedure The p as address Crane Node (q) P a- q End procedure	End procedure Procedure mert Node (*p.: address, what: inforge): If p is NULL Thum Crane Node (p) Anto(p) of miles Next(p) of NULL Else Quelone of as address Crane Node(a) Into (p) of vila Next(q) of p P of a Find piecedure	End procedure Procedure merch Node (*p. address, most inforgee). If p is NULL Then: Creane Node (p) Ango (p) a- node Else Creane Node (a) Ingo (p) a- Node (a) Next (a) a posedure Find procedure	End procedure Procedure mert Node (*p. address, what inforges). If p is NULL Thum Crease Node (p) Info (p) of nilos Mext(p) of Node Else Crease Node (q) Node (q) Node (q) Page (q) of plan Node (q) Page (q) of plan Node (q) Find procedure	End procedure Procedure mert Node (*p.: address, what: inforge): If p is NULL Thum Crane Node (p) Meso(p) of miles Mext(p) of NULL Else Declare of as address Crane Node(of) Mext(of) of NULL Else Pactor of as address Crane Node(of) Next(of) of NULL End procedure Fund procedure	End procedury End procedury Procedure mert Node (*p.: address. what: interpope): If p is NULL Thuri Create Node (p) Info(p) of nilar Mext(p) of NULL Else Peclare of as address Create Node(of) Info(p) of NULL End procedure Find procedure
End procedure Procedure mert Node (*p. address, what interpo). If P is NULL Them Creak—Node (p) Inspo(p) of min Mext(p) of NULL Else Declare 9 as address Creak—Node (q) Inspo(9) of wha Next(q) of precedure Fud precedure	End procedure Procedure mert Node (*p. address, what interpose). If P is NULL Them Create—Node (p) Info(p) of min Mext(p) of NULL Else Pectore of as address Create—Node (a) Info(p) of vila Next(q) of peredure Find procedure	End procedure Procedure mert Node (*p. address, what: inforgee). If p is NULL Then: Creak - Node (p) Ango (p) a- Node (p) Rectore of as address Creak - Node (d) Ingo (g) a- vila Next (g) a- p Pa-q End procedure	End procedure Procedure mert Node (*p. address, what interspe): If p is NULL Them Crane Node (p) Ango (p) of min Mext (p) of Nole Else Declare of as address Crane Node (q) Next (q) of vila Next (q) of powedure Find procedure	End procedure Procedure merry Node (*p.: address, what inforge): If p is NULL Thum Crase Node (p) Mes(p) of miles Mext(p) of NULL Else Crase Node(q) Impo (g) of value Next(q) of p Find procedure That if The procedure	End procedure Procedure meet Node (*p. address, what inforges). If p is NULL Them Create—Node (p) Info(p) q— who Mext(p) q— Node (p) Else Declare q as address Create—Node (q) Info(q) q— vila Next(q) q— p Find precedure	End procedure Procedure mert Node (*p.: address, what: inforge): If p is NULL Them Create Node (p) Info(p) of miles Next(p) of NULL Else Declare of as address Create Node (q) Info(p) of vala Next(q) of precedure Find precedure
End procedure Procedure near+Node (*p: address, what interpos): If p is NULL Then: Create—Node (p) Insto(p) a— Noll Else Declare a as address Create—Node (q) Insto(q) a— Noll Not(q) a— p P a— q Find precedure	End procedure Procedure mert Node (*p: address, what inforge): If P is NULL Then: Creak Node (p) Inso(p) of min Mext(p) of NULL Else Declare of as address Creak Node (q) Inso(q) of what Node (q) Por q For q For q Indo procedure	End procedure Procedure mert Nocke (*p: address, what inforge): If p is NULL Then: Create—Nocke (p) Anso (p) of min Mext(p) of NULL Else Declove of as address Create—Nocke (q) Impo (q) of what Next(q) of p Por q End procedure	End procedure Procedure mert Nock (*p: address, what inforge): If p is NULL Then: Crease Nock (p) Anso(p) of nilos Mext(p) of NULL Else Declare of as address Crease Nock (a) Inso(p) of what Nock (a) For a For a For a Tool if End procedure	End procedure Procedure merit Node (*p.: address, what: inforge): If p is NULL Thum Crane Node (p) Info(p) a nilar Next(p) a Node (p) Else Crosse Node (q) Impo(q) a volar Next(q) a p Find procedure Find procedure	End procedure Procedure mert Node (*p.: address. what: inforge): If p is NULL Them: Crank - Node (p) Info(p) of miles Mext(p) of NULL Else Declare q as address Crank - Node(q) Info(p) of NULL Else Parallel procedure Find procedure Find procedure	End procedure Procedure mert Node (*p: address, what: interpoe): If p is NULL Trum: Creak - Node (p) Info(p) of miles Else Rectore of as address Creak - Node (q) Info(p) of wha Next(q) of p P of q Find presedure
Procedure mert Node (*p. address, what interpo). If P is NULL Them Creak - Node (p) Inspo(p) of min Mext(p) of NULL Else Reclare 9 as address Creak - Node (q) Inspo(9) of wha Next(q) of powedown Fud if End procedure	Procedure insert Node (*p. address, what interged). If P is NULL Then! Creak - Node (p) Anso(p) - node Next(p) - node Else Declare 1 as address Creak - Node (a) Inso(p) - vola Next(q) - p P - q End procedure	Procedure mert Node (*p: address, what inforget): If p is NULL Them: Create—Node (p) Inspo(p) of miles Next(p) of NULL Else Declare 1 as address Create—Node (q) Inspo(p) of what Next(q) of p Por q Ford if End precedure	Procedure mert Node (*p: address, what inforge). If P is NULL Them: Create—Node (p) Inso(p) of node Mext(p) of NULL Else Declare of as address Create—Node(d) Inso(g) of what Next(q) of p P of q End procedure The	Procedure mert Node (* p. address. what interpolice for it NULL Them. Creak - Node (p) Inso(p) of nilar Mext(p) of NULL Else Peclare of as address Creak - Node (q) Inso(q) of Nular Next(q) of procedure Find procedure	Procedure mert Node (*p. address, what interspe). If P is NULL Them Creak Node (p) Inso(p) of nilar Mext(p) of NULL Else Pectore of as address Creak Node (a) Inso(p) of vilar Node (a) Inso(p) of vilar Node (a) Inso(p) of position of a distribution of a distribution of a second of a distribution of	Procedure mert Node (*p. address, what inforge). If p is NULL Them Creak Node (p) Info(p) of milan Next(p) of NULL Else Declare of as address Creak Node (q) Info(p) of vilan Next(q) of procedure Find procedure Find procedure
Procedure merry Node (*p: address, what inforget); If p is NULL Then: Creak - Node (p) Inso(p) of miles Mext(p) of NULL Else Declare of as address Creak - Node (g) Inso(p) of vilas Next(g) of power of	Provider merry Node (*p: address, what inforget). If p is NULL Them Creak - Node (p) Inspo(p) of node Else Declare of as address Creak - Node (q) Inspo(q) of volun Next(q) of p For of The content of the	Provider mert Node (*P: address, what inforge): If P is NULL Them: Create - Node (P) Info(P) of miles Mext(P) of NULL Else Declare of as address Create - Node (q) Info(P) of what Next(q) of p Por q End providence	Procedure near+Node (*P: address, what inforge); If P is NULL Them: Creak - Node (P) Info(P) - Node (P) Next(P) - Node (A) Info(P) - Node (A) Info(P) - Node (A) Info(P) - Node (A) End procedure Find procedure	Provide insert Node (*P: address, what inforge): If P is NULL Them: Creak - Node (P) Inspo(P) - Node (P) Else Declare 9 as address Creak - Node (9) Node (9) - War Node (9) - War Node (9) - P Pag Find precedent	Procedure merry Node (*p: address, what inforge): If p is NULL Then: Create—Node (p) Insto(p) of miles Next(p) of NULL Else Declare of as address Creat Node (q) Insto(q) of what Next(q) of p Paragraphic Find precedure	Propane mert Node (*p: address, what inforge): If p is Nucl Then: Crack - Node (p) Anso (p) of min Mext(p) of Nucl Else Reclare of as address Crack - Node (q) Inso (p) of vila Node (q) Fod if End piecedure
Properties mert Node (*p. address, what: informed): If p is NULL Them: Create Node (p) Medip) of nilar Mext(p) of NULL Else Create Node (q) Impo (g) of what (nilar) Node (q) Impo (g) of what (nilar) Node (q) For q End procedure	Proparine mert Node (*p: address, what: informed). If p is NULL Them: Create - Node (p) Mext(p) of miles Mext(p) of miles Else Pectore of as address Create - Node (d) Into (p) of what Next(q) of what Next(q) of p Find piecedine	Procedure merry Node (*p: address, what: inforge): If p is NULL Them: Create—Node (p) Anso(p) of node Else Declare of as address Create—Node (g) Inso(p) of vola Node (g) For g For g	Procedure merry Node (*p: address, what inforge): If P is NULL Them: Create—Node (p) Inso(p) of miles Mext(p) of niles Else Declare a as address Create—Node (a) Inso(p) of wiles Next(a) of powedure For a g Fo	Procedure insert Node (*p: address, what inforger): If P is NULL Them: Create—Node (p) Info(p) of node Else Declare q as address Create—Node (q) Info(q) of vila Next(q) of precedure Fund precedure That precedure	Procedure insert Node (*p. address, what inforge): If p is NULL Them: Creak - Node (p) Anso(p) of node Mext(p) of NULL Else Declare of as address Creak - Node (g) Insp (g) of volar Node (g) For g For g	Proparie insert Node (*p. address, what inforge): If P is NULL Then: Create—Node (p) Inspo(p) of node Else Declare q as address Create—Node(q) Inspo(q) of vila Next(q) of p P of q Find procedure
Procedure mert Node (*p. address, what informed). If p is Nouth Them: Create—Node (p) Info(p) of miles Else Rectore of as address Create—Node (g) Info(p) of what Next(g) of what Next(g) of p P of g Fud if End piecedure	Procedure mert Node (*p. address, what inforge). If p is NULL Them: Create - Node (p) Into (p) a- NULL Else Declare a as address Create - Node (a) Into (p) a- Null Next(a) a- p Find precedure The precedure In a	Produce mert Node (*p: address, what: inforge): If p is NULL Them: Crost - Node (p) Info(p) of minimal Mext(p) of Null Else Declare of as address Crost - Node (q) Info(p) of what Node (p) Red if End procedure	Produce mert Node (*p: address, what: inforge). If p is NULL Them: Create Node (p) Ango(p) of milan Mext(p) of Null Else Declare A as address Create Node(q) Ingo(q) of what Next(q) of p For q Fo	Provide mert Node (*p: address, what inforge): If P is NULL Then: Create - Node (p) Inspo(p) of nilar Mext(p) of NULL Else Declare of as address Create - Node (d) Inspo(q) of vilar Next(q) of powedows For of For	Procedure merry Node (*p: address, what interspe). If p is NULL Them: Creak - Node (p) Inspo (p) a- NULL Else Declare a as address Creak - Node (a) Inspo (a) a- NULL End if End precedure	Propare merr Node (*p: address, what inforge): If p is NULL Then: Create - Node (p) Anso(p) of node Else Declare of as address Create - Node (g) Inso(g) of vila Node (g) Find presedure
Procedure mert Node (*p. address, what informed): If p is NULL Them: Create Node (p) Medip) of nilar Mext(p) of NULL Else Create Node (g) Impo (g) of what Null Node (g) For g End procedure	Procedure meet Nock (*p. address, what informed). If p is NULL Them Create Nock (p) Ango(p) of miles Mext(p) of miles Else Declare of as address Create Nock (q) Impo (g) of what Next(q) of p For of For o	Procedure meet Node (*p. address, what informed). If p is NULL Them Create Node (p) Ango (p) of mini Else Rectare of as address Create Node (q) Ingo (q) of mini Next(q) of mini Por q Find piocedure The procedure The pro	Problem ment Node (*p. address, what informed). If p is NULL Them: Creak Node (p) Anso(p) of nilon Mext(p) of NULL Else Creak Node (q) Inso (g) of Null Node (q) Inso (g) of Null Node (q) Porgan Porgan Find if End if End procedure	Procedure near Nock (*p. address, what informed). If p is NULL Them: Create - Nock (p) Med (p) Mext(p) of nilon Next(p) of NULL Else Create - Node (q) Into (g) of what Node (q) Next(q) of p Find procedure	Procedure mert Node (*p. address, what informed). If p is NULL Them: Create - Node (p) Info(p) of nilar Mext(p) of nilar Mext(p) of Null Else Declare A as address Create - Node (g) Impo(g) of vilar Note(g) of plan Fod if End piecedure	Probably insert Node (*p: address, what inserge): If p is NULL Then: Creak - Node (p) Insp(p) a- node Else Declare a as address Creak - Node (a) Insp(p) a- vila Next(a) a- p Fad is End procedure
Lese Else Reclave of as address Crease - Nodela) Info (9) & More Nortal (9) & More Find prevedure	Lesse Nort Then: Create Node (p) Ango (p) of miles Else Reclave of as address Create Node (q) Ingo (p) of miles Next (q) of p Por q Ford if End prevedure	The pris Nucl Them. Charle - Node (p) Ango(p) a- Nolu Else Declare 9 as address Charle - Node(q) Ingo(q) a- Nolu Next(q) a- p Pa-q Find procedure	The pris Nucl Them: Crose - Node (p) Ango(p) of miles Mext(p) of Node Else Crose - Node(q) Ingo (q) of miles Next(q) of plan Find procedure Find procedure	The pris Nucl Them: Create - Node (p) Into (p) and node Else Declare as address Create Node (a) Into (p) and node Next(a) and node Page 14 Find precedure	The pris Nucl Then: Create Node (p) Into (p) of miles Rector of as address Create Node (d) Into (p) of miles Create Node (d) Next (q) of miles Next (q) of miles Find procedure	If P is NULL Then: Create—Node (P) Info(P) &— node Else Declare of as address Create—Node (g) Info(g) &— vilar Next(g) &— vilar Next(g) &— P Fa- g End precedence
The pris Nucl Trum Create - Node (p) Anso (p) of minor Mext (p) of minor Else Declare of as adobuss Create - Node (q) Insp (q) of minor Next (q) of p Find procedure	the pis Nucl Them Create - Node (p) Anso(p) of minimal Mexit(p) of Nucl Else Rectione of as address Create - Node (g) Impo(g) of minimal Mexit(g) of poundance Find previouse Find if End of poundance I have been a second of the minimal mexicolor of the minimal m	The p is Nort Then: Create - Node (p) Areo(p) of minimal Mexit (p) of North Else Create - Node (q) Into (p) of Mode (q) Into (p) of minimal Mexit (q) of point	The pis Nout Them: Create - Node (p) Inspo(p) of minor Else Create - Node (q) Inspo(p) of Nout Inspo(p) of minor Next(q) of p Pod if End presedure	The pis NULL Them Create Node (p) Inter(p) of nilar Else Rectore of as address Create Node (d) Inter(p) of vilar Next(q) of vilar Next(q) of powdure Fud procedure	The p is NULL Them: Charle Node (p) Mext(p) of nilw Mext(p) of NULL Else Charle Node (q) Inspo (g) of Null Node (q) Por q Find if Find procedure	The p is NULL Then: Create - Node (p) Info(p) & nilar Mext(p) & Node (a) Info(p) & Node (a) Info(p) & Node (a) Fod if End posedure
Create Node (p) Intro(p) of miles Else Rectave of as address Create Node (g) Intro(g) of what Next(g) of p Page Find prevedure	Create Node (p) Inpo(p) of miles Else Declare of as address Create Node (g) Inpo(g) of wha Next(g) of p P of g End if End piecedare	Create Node (P) Ango (P) of miles Mext(P) of Note Else Declare of as address Create Node (g) Into (g) of miles Node (g) Fod of Fod of	Create - Node (P) Ango(P) of miles Mext(P) of Miles Else Declare of as address Create - Node (a) Ingo(p) of miles Create - Node (a) Ingo(p) of miles And prevedence Frag if End prevedence The pre	Create Node (P) Inspo(P) of Mon Next(P) of Mon Else Reclove of as address Create Node (a) Inspo(p) of Mon Next(q) of P For of	Else Rectore of as address Create Node (a) Into (p) and Note Else Auto (p) and Note Into (p) and Note Auto (p) and Note Fra p	Else Declare of as address Create Node (a) Info (a) of Mula Mexical it End precedure The p
Create - Node (p) Inter(p) of nilos Else Rectore of as adobuss Create - Node (g) Inter(g) of vilas Next(g) of p Find procedure	Create - Node (P) Info(P) a- nion Mext(P) a- Note Else Declare a as address Crease - Node(a) Info(p) a- nion Next(a) a- p Pa-a Find prevedence Tool if The prevedence	Creak - Node (P) INFO(P) - MIN Else Declare 9 as address Creak - Node (9) Info(9) + War Next(9) + War Next(9) + P For 9 For 9 For 9 For 14 Find prevedure	Create - Node (p) Info(p) of minimal Mexist (p) of Note (p) of Note (p) Else Create - Node (q) Info(p) of minimal Mexist (p) For of presedure End presedure	Create Node (p) Into (p) of minimal Mext (p) of Note Else Create Node (q) Into (q) of vilan Not (q) of p Find procedure Find procedure Office (p) of minimal positions of the positions o	Create Node (p) Areo(p) of now Next(p) of Nove Else Crease Node(q) Into (q) of Node(q) Next(q) of P Porg Find piecedure	Create Node (p) Into(p) of now Else Reclaim of as address Create Node (d) Into(p) of wha Next(q) of p Find presedure Find presedure
Ango(p) of minimal Mexit (p) of North Else Declare of as adobuss Crease Node(q) Impo(q) of what Nexit(q) of p Por q Find presedure	Augo (p) of miles Else Rectore of as address Crease Node (g) Impo (g) of miles Next (g) of power of the	Auto(p) a- Morri Else Reclave 9 as address Corase Node(9) Inspo(9) a- Morri Next(9) a- Morri Pa-9 Ford procedure	AMERICAN A AS Address Create Node (4) Inpo (9) 4 Node (9) Impo (9) 4 Node (9) Next(9) 4 P Page Find procedure	Anso(p) of min Mext(p) of Note Else Declare of as address Crease Node(g) Inso(g) of what Next(g) of p p of g End it End piecedare	Ango(p) of miles Else Rection of as address Consult Node(a) Inpo(q) of miles Node(q) of miles For a	Auto(p) of miles Else Declare of as address Crease Node(a) Into(p) of miles Auti(a) of Node(a) Por of Find previouse Find previouse The previouse The previouse The previouse The previous of the
Mext(p) of Now Else Declare of as address Crosse Node(a) Inpo(a) of Node Node(a) Por of Por of Portor	Auto(p) of More Else Declare of as address Crosse Node(q) Into (1) of Mar Next(q) of Mar Porq Porq Ford if End procedure	Mext(p) of More Else Declare of as address Crosse Node(q) Into (q) of Mar Node(q) For of For of	Auto(p) of Mon Else Reclave of as address Corase Node(q) Impo(q) of Maria Next(q) of P Find precedure	Augo(p) of min Else Reclave of as address Crease Node(a) Impo(a) of vila Next(a) of P Pag End procedure	Mext(p) of now Next(p) of Nour Else Declare of as address Crosse Node(q) Insp (9) of wha Next(q) of p For a For a	Mext(p) of now Else Reclave of as address Consile Node(a) Insp (a) of volan Node(a) of posedure Ford procedure Find procedure
Else Declare 9 as address Crean Node (9) Impo (9) 4- Mar Node (9) For 9 F	Else Declare of as address Crease Node (g) Info (g) of Mar Next (g) of Mar Next (g) of P	Else Rectore a as address Crease Node(a) Inspectation Next(a) = P Page End procedure	Else Reclave 9 as address Crear Node (9) to Node (9) Into (9) to Node (9) Next (9) to P Page Find procedure	Else Reclave 9 as address Crosse Node (9) Impo (9) 4- War Next (9) 4- P P 4-9 End procedure	Else Reclave 9 as address Crease Node(9) Impo(9) 4- Was Nest(9) 4- Was Nest(9) 4- P Fa-9 Fa-9 Ford procedure	Else Declare of as address Compile - Modeld) Info (9) 4- War Next (9) 4- War Next (9) 4- P P 4- 9 Find previouse
Else Declare 9 as address Crosse Node (9) Inpo (9) 4 what Next (9) 4 P End presedure Fa-9 End presedure	Else Declare 9 as address Crean Node (9) Info (9) 4- volan Next(9) 4- p Fa-9 Ford 14 End 14 End 14 End 14	Else Declare of as address Cranse Node (a) Into (a) the roland Node (a) the procedure End it for a formation of the procedure That procedure	Else Rectore 9 as address Crosse Node (9) Into (9) 4 War Next (9) 4 P For 9 For	Else Reclave 9 as address Crosse Node(9) Into (9) to what Next(9) to P Fa-9 For 9 For	Else Rectore 9 as address Crear Node(9) Impo(9)4 whan Next(9)4 P Find precedure Find precedure	Else Reclave 9 as address Crear Node(9) Impo(9) + Made(9) Node(9) + Ma Page 14 Fag 14 End procedure
Else Declare 9 as address Crosse - Mode (9) Impo (9) 4 - War Next(9) 4 - P Pag End procedure	Else Declare 9 as address Crosse Node (9) Info (9) 4 volar Next(9) 4 P Pag Find procedure	Else Declare 9 as address Crease Node (9) Inpo (9) 4- vular Next(9) 4- p P 4- 9 End prevedure	Else Reclave of as address Crosse Node (a) Impo (a) 4- value Next(a) 4- p P 4- a Find precedure	Else Declare of as address Crosse Node (a) Info (9) 4- volun Next(9) 4- p Page Find provident	Else Reclave 9 as address Crease Node (9) Impo (9) 4- Wlan Next (9) 4- P Ford presedure Find presedure	Else Peclore of as address Crear Node(a) Impo(a) 4 mla Node(a) 4 P For a
Else Rectave of as address Crean Node (g) Inpo (g) 4- vila Node (g) Fa- g End piecedire End piecedire	Else Reclam 9 as address Crean Node (9) Impo (9) 4 vila Nexi(9) 4 P P 4 9 End piecedure	Else Reclove of as address Crear Nodela) Impolit For a For a	Else Reclare 9 as address Create Node (9) Impo (9) 4- vila Next (9) 4- P Far 9 Far 9 Find prevedure	Else Reclare 9 as address Crease Node (9) Impo (9) 4- Man Node (9) Pa-9 Food it End it End presedure	Else Reclave 9 as address Crease Node (9) Inpo (9) 4- Man Next(9) 4- P Fad if End presedure	Else Rectore 9 as address Correct Node (9) Impo (9) 4- Man Next (9) 4- P For 9 For 9 For 9 For 14 End prevedure
Else Declare of as address Crear Node (9) Impo (9) 4- Mar Nex 1(9) 4- P Pa-9 Find procedure	Else Creak As address Creak Nodela) Info(9) & Wala Next(9) & P Fa-a Find piecedire The pie	Else Declare of as address Crear Nodela) Impolator Impolator Nextagle P Page Find presedure That presedure	Esse Creax of as address Creax Node (a) Impo (9) 4 - vila Nex (a) 4 - p Page 1 - p Find procedure	Erse Rectave 9 as address Crease Node (9) Impo (9) 4— Man Next(9) 4— P P 4— 9 Find prevedure	Evelore 9 as address Crease Node (9) Inpo (9) 4 vila Node (9) 4 vila Node (9) 4 vila Page 14 End presedure	Else Declare 9 as address Creax Node (9) Impo (9) 4 rolar Next (9) 4 P Fa-9 Frage Frag
Reclave of as address Correct Node (a) Inpo (9) 4- vola Next (a) 4- P For a For a	Reclave 9 as address Chesic Node (9) Inpo (9) 4- vila Next(9) 4- P Fa-9 Ford if End procedure	Peclare of as address Creare Node (a) Impo (a) & vila Next(a) & P For a For it End previouse	Declare of as address Creare Node (a) Impo (9) 4- vila Next (a) 4- P For a For a For a Find prevedure	Peclare of as address Creax Node (a) Impo (1) 4- vila Nex (a) 4- p For a precedure Find precedure	Evelipe of as address Crean Node (9) Inpo (9) & vila Next(9) & P For 9 For 9 For 14 End procedure	Reclave of as address Corone Node (a) Impo (a) 4 May Next (a) 4 P For a Fo
End procedure End it End it End it End procedure	End procedure End procedure End procedure	End precedure End precedure The pr	End procedure End procedure Find procedure Tool of the colours Tool	End procedure End if End if End procedure	End procedure End procedure Find procedure	Evalia Poudus Fra poudure For a sadius Audit For a sadius For a sad
Cronk Node(9) [npo(9) + nla Node(9) Fad if End procedure	Crone Node(9) (npo(9) + wla Next(9) + P Ford procedure	End piecedure End piecedure	Cross Node(9) Inpo(9) + vila Next(9) + P For 9 For 9 For 9 For 19 For 1	Crosse Node(9) Inpo(9) + vila Ford it Ford it The procedure	Cross Node(9) (npo(9) + vila Node(9) + p Fa-9 For 9 F	Cross Node(9) Inpo(9) + Wa Node(9) + P Fad procedure
End presedure End if End if End of the point of the po	End procedure Find if Find if Find if Find if The comment of	Cross Node (9) + who Node (9) + who Node (9) + who Ford procedure	End procedure Find procedure The p	End piecedure Find it Find precedure Corone Node (4)	Cross Node(9) Node(9) + who For 9 For 9 For 9	End it End it The preduce I have a presedure
Info(9) + Mail	Info(9) + Max Next(9) + P Find presedure	End procedure Find procedure Find procedure	End proudure End if End if End proudure	End procedure Find procedure	End piecedure End it End it	End procedure Find procedure
End procedure End procedure	End piecedure End piecedure	End procedure Find procedure	End procedure Find procedure	End proudure End if End proudure	End procedure Find if Find procedure	End pendure End it End of P
End procedure	End proudure	End procedure	End procedure	End procedure	End prevadure	End procedure
End presedure	End proudure	End procedure				
End presedure	End procedure	End proudure	End piecedire	End procedure	End pieudine	End procedure
End procedure	End procedure	End proudure	End president	End piecedine	End proudure	End proudure
End peredure	End presedure	End precedure	End procedure	End president	End procedure	End proudure
End piecedire	End poudure	End proudure	End peridure	End piecedure	End procedure	End piecedure
End procedure	End presedure	End procedure	End proudure	End pieudine	End pionedine	End pioudine
End procedure	End proudure	End piecedure	End piecedure	End piecedure	End piecedure	End procedure
		The production of the producti	PARTIE DE LA PROCESSION	The products of	PARTICIPATION OF THE PROPERTY	The state of the s
		PARTIE IN	** TATALE ***	NUMBER OF THE PARTY OF THE PART		**************************************
		**************************************	PARTY IN	Market and	- TOREST	1005 1.1
		PARTY IN	** TOTAL ***	PARTY	PARTY OF	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
		PARTIE IN	** TODAY ***	Mark of the same o	Mark on	100 mm 10
		PARTIE IN	PARTY IN	NAME OF THE PARTY	- TOOL	
		PARTIE IN	PARTY IN	Market and	**************************************	NUMBER OF THE PARTY OF THE PART
		PARTIE IN	PARTY IN TRACT	PARTY AND DESCRIPTION OF THE PARTY OF THE PA	PARTY OF TAXABLE	Name of the latest device of t
		PARTIE IN	PARTY IN TRACE	NAME OF TAXABLE PARTY.	- TORE	N 2005
		** TADEL!	** TADEL!	Market and	- TOUR!	NO. 1.
		PARTY IN	PARTY IN TRACE	- TEDELS	PARTY AND THE PROPERTY OF TAXABLE PARTY	Name of the state
		PARTICIAL IN	- Trout	Maria	" TOOL"	1 TOTAL
		* TADEL!	PARTIE IN TRACE	- Troute	- TODGE	NUMBER IN
		PARTIE IN	" TIDEL"	PARTY AND DESCRIPTION OF TAXABLE	"POCE" IN	Name of the state
		*ADOLT	* TORE	NAMES AND ADDRESS OF THE PARTY	- TOOLS	19,000 m.
		* TADEL	** 33057	India	MARCHE AND THE PROPERTY	National and a state of the sta
	TOTAL STREET	PARTICIAL DESCRIPTION OF THE PROPERTY OF THE P	" IXDEX. "	MADELE	" TORE"	NACE
	NOTE IN COLUMN TO SERVICE AND ADDRESS OF THE PARTY OF THE	** India	MADEL 144	India	PARTIE AND TAXABLE PARTIES	** TADEL ***
	NIDOTE	*ADOLT ***	" IIDEL"	MADELE	"place" and	Name of States
	NOTE	**BOIL ***	" INDEED.	"IDOLL ""	" TEDELS"	NUMBER
	WINDLE	*ADOLT ***	"ADOLE	" INDEE!	Marie	NAMES OF TAXABLE
	W.DOLL	** IIII	" TIDEL "	MARCE	"place" and	NO STORE
The same of the sa	WINDER AND	" TIDEL	MARKET AND THE PERSON NAMED IN COLUMN NAMED IN	NIBOLE AND	MARGET AND	NUMBER

2) Crease Node (2) Function create Node (x: intotype) - baddiess Aluc memory for p with Size of FlintList Into(p) 4-x Nex+(P) + NULL Return P End Function (10 2025.03.09 22:55 z). Crease node (1) Procedure Crease-Node (*P: by rex address): Alloc memory for P with size of Elm+List Mext (P) 4 NULL If p is not NULL Then: Pun Fnd IF End procedure