(3x.xa) $(3x.xa) \Rightarrow (3y.yy)(3x.xa) \Rightarrow (3x.xa)(3x.xa) \Rightarrow$:3 5000 ((yx.ya.xa)(ya.a) => (xxx) => (xyx) => (xyx) => (xyx) => (xyx) => w $(3x.y)((3y.yy)(3x.xxx)) \Rightarrow (3x.y)((3x.xxx)(3x.xxx)(3x.xxx))$ 27: (3x. x x) 3 1801' 80 bi call by value of onno Projic 3x 724' 25 82 rec Fact 2 > fact (rec fact 2) => (2F. 2n. if n=1 then 1 else n*fn-1) (rectate)(3) >> > > n. if n=1 then 1 else n*rec fact n-1) 2 => if 2=1 then 1 else 2* rec fact 2! ⇒ 2 * rec Fact 1 ⇒ 2 * Fact (rec fact 1) ⇒ 2* () F. An, if n=1 then 1 else n*Fn-1) (rec Fact 1) => 9* (An. if n=1 then 1 else n* rec fact n-1)1 => 2* if 1=1 then 1 else 1* recFact 1-1=) 2*1=2 = 2 2/160 test (or try Fls) ab => > (al. am. an. Imn)((ab. ac.btac)(at.x.t)(at. af. f)) ab)> (71.2m. 2n. 1 mn) ((7t. 26.t)(2t. 26.t)(2t. 26.t)) ab = (21.7m. 7n. |mn) ((1t. 7F.t) ab =) (7t. 7F.t) ab =) a

not = 76.6 Fls tru . Of = 76.7c. b truc -e yos 0 Dor = >x·yy·(not (or x y)) => xx·yy ((xb.b Fistin) ((xb.xc·b tin c) xy)) >> >x. >y ((xb.b Els tru) (x tru y)) > xx. >y ((x tru y) Fls tru) nor tru FIS >> >x. >y (1x tru y) FIS tru) tru FIS >> >y. |(tru tru y) FIS tru) FIS @ > (try try fls) fls try => ((xt. xx.t)(xt. xc.t)(xt. xc.f)) fls try => ((AF. (At. AF. +)) (At. AF. F)) FIS try => (At. AF. F) FIS try =) nor try try > ax. ay ((x try) FIS tru) try try => (ay. (try try y) FIS try) try => (tru tru tru) FIS tru => ((x+.x.+)(x+.x+.+) (x+.x-.+) (x+.x-.+) (x+.x-.+) → (() F. (>+. >F. +)) (>+. >F. F) (>+. >F. F) (>+. >F. F) (x 76.5)(x+. xe.6)(x+.x-1) => x6. ((x+.x6.6)(x+.x6.4)) => x+. x6.6 => Fls :2 aske Succ C1 = (710.75.77.5(n57)) C1 => 75.77.5(C157) Succ C1 = (70, 75, 77, 5 (n 57)) C1 => 75, 77, 5 (C, 57) were or by that the certifier and stabolder cult by salle of the · shed for very bigy iseven = 7x. (x 176.6 Fls tru) | tru) = 7x. (x not tiu) :772) is even C3 => (xx. (x not tru)) C3 -> C3 not tru => (75.72.5(5(52)) not tru (3) -> 72. not (not (not 2)) try -> not (not (not (ru))) -> not (176.6 FIS tru) (176.6 FIS tru) tru) => not (176.6 FIS tru) (17.74.4) FIS tru) => not (126.6 Fls tru) Fls) => not (Fls Fls tru) =) (26.6 Flstru) (Fls Fls tru) = (20.0 Fls tru) tru => tru fls tru => (2t. 2f.t) Fls tru => Fls

iseven cy = (7x.(x not tru)) cy = cy not try => (75.72.5(5(5 2))) not try	
not (not (not (not z)))) try > not (not (not try))) =>	
Not (not (not (sign))) try => not (not (not (not try)))	
Not(not(ris))) => not(not(tru)) => not(ris) => tru	1
: Y DKe	
: any T-Bod Du T	0
i Alen H Leich	
T- true	
FiBool > Bool + true: Bool	
T-true T-False	3
2 1 2 1 2 1	F-if
	TA
F:Bool → Bool + F:Bool → Bool + (if truether false else true): Bool	
F:Bool - Bool + F:Bool - Bool + F(if true then False else true): Bool F: Bool - Bool + F(if true then False else true)): Bool	
F:Bool-Bool F:Bool-Bool F:Bool-Bool (if truethen Falce else true): Bool F: Bool-Bool F (if true then False else true)): Bool my 4 h nog T=Bool-Bool	3
F:Bool-Bool Ny (
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true)): Bool : might by night T=Bool > Bool T-var x: Bool > Bool x: Bool x	3.
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true)): Bool True X: Bool > Bool + Bool - Bool - Bool - Bool - True - X: Bool - Bool - X: Bool - Bool - F:Bool > Bool - F:B	3.
F:Bool > Bool + F:Bool > Bool	3 The
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true): Bool F: Bool > Bool T=Bool > Bool T-year X: Bool E: Bool > Bool F: Bool > Bool F: Bool > Bool X: Bool F: Bool > Bool F: Bool > Bool X: Bool F: Bool > Bool F: Bool > Bool X: Bool F: Strue : Bool F: Bool > Bool X: Bool F: Strue : Bool F: Bool > Bool X: Bool F: Strue : Bool F: Bool > Bool X: Bool F: Strue : Bool F: Bool > Bool X: Bool F: Strue : Bool F: Bool > Bool X: Bool F: Strue : Bool F: Bool > Bool X: Bool F: Strue : Bool F: Bool > Bool F: Bool > Bool F: Strue : Bool F: Bool > Bool F: Strue : Bool F: Bool > Bool F: Strue : Bool F: Strue : Bool F: Bool > Bool F: Bool > Bool F: Strue : Bool F: Bool > Bool F: Strue : Bool F: Bool > Bool F: Bool > Bool F: Bool > Bool F: Strue : Bool F: Bool > Bool F: Boo	orue of Inthe Book
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + F(if true then False else true)): Bool F:Bool > Bool + F(if true then False else true)): Bool True X:Bool = Bool Deel 7: K:Bool Deel 7: K:Bool Deel 7: Bool > Bool F:Bool >	orue of the Boot
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + Bool - Bool - Bool T-year X: Bool = Bool - Bool - Bool - F:Bool > Boo	orue of Inthe Book
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F: Bool > Bool + F(if true then False else true)): Bool F: Bool > Bool + F(if true then False else true)): Bool True X:Bool T - False T - False T - False X:Bool X:Bool X:Bool X:Bool X:Bool X:Bool E:Bool > Bool F:Bool > Bool X:Bool F:Bool > Bool X:Bool F:Bool > Bool X:Bool F:Bool > Bool X:Bool Filse False Rool > Bool F:Bool > Bool Tiff Then False else true): Bool F:Bool > Bool F:Bool > Bool F(iff Then False else true): Bool F:Bool > Bool F(iff Then False else true): Bool F:Bool > Bool F(iff Then False else true): Bool F:Bool > Bool F(iff Then False else true): Bool F:Bool > Bool F(iff Then False else true): Bool F(iff Then False else True)): Bool > Bool True True	orue or the Boot T-APP
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + F(if true then False else true)): Bool F:Bool > Bool + F(if true then False else true)): Bool True X:Bool = Bool - Bool - Bool Null True X:Bool = Bool - Bool - Bool X:Bool X:Bool X:Bool Bool > Bool - Bool - Bool - Bool - Bool F:Bool > Bool - F:Bool > Bool F:Bool > Bool E:Bool > Bool F:Bool > Bool	orue or the Boot T-APP
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + Bool - Bool - Bool T-ver X:Bool - Bool - Bool - F:Bool > Bool - F:Bool - Bool - F:Bool > Bool - F:Bool > Bool - F:Bool - - F:Boo	T-APP
F:Bool > Bool + F:Bool > Bool > Bool + F:Bool + F:Bool > Bool + F:Bool + F:Bool + F:Bool + F:B	orue or the Boot T-APP
F:Bool > Bool + F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + F(if true then False else true): Bool F:Bool > Bool + F(if true then False else true): Bool T-year (Rool = If:Bool > Bool - Bool - Bool - Bool - Bool - Rool - Bool - F:Bool > Bool - F(if x then False else true): Bool F:Bool > Bool + F(if x:Bool - F(if x then False else true)): Bool > Bool - F:Bool > Bool - F:Bool > Bool - F(if x then False else true)): Bool > Bool - F:Bool > Bool - F(if x:Bool - F(if x then False else true)): Bool > Bool - F:Bool > Bool - F:Bool > Bool - F(if x then False else true)): Bool > Bool - F:Bool > Bool - F:Bool > Bool - F(if x then False else true)): Bool > Bool - F:Bool > Bool - F:Bool > Bool - F(if x then False else true)): Bool > Bool - F:Bool > Bool > Bool > Bool - F:Bool > Bool	orue or the Boot T-APP
F: Bool > Bool F: Fire then False else true!): Bool F: Bool > Bool F (if true then False else true!): Bool True X Bool = Gool T - False T	TYAY
F:Bool > Bool + F:Bool > Bool > Bool + F:Bool > Bool > F:Bool > Bool > F:Bool > F:Bool > Bool > F:Bool > F:B	Tray
F:Bool > Bool + F:Bool > Bool + Fife true then False else true): Bool F: Bool > Bool + Fife true then False else true): Bool F: Bool > Bool + Fife true then False else true): Bool T-False T-ture **(Rool = Still + Bool > Bool - Dull) T-False T-ture **(Rool = Still + Bool > Bool - Dull) T-False T-ture **(Rool = Still + Bool > Bool - Fiber + False else true): Bool F: Bool > Bool + File + Bool + Fiber + False else true): Bool F: Bool > Bool + File + False else true): Bool F: Bool > Bool + Fiber + False else true): Bool F: Bool > Bool + Fiber + False else else true): Bool > Bool T-False T-ture **(Rool + Bool + Fiber + False else + False +	TYAY
F:Bool > Bool + F:Bool > Bool + File + Folse else true!: Bool F:Bool > Bool + File + File + Folse else true!: Bool F:Bool > Bool + File + File + Folse else true!: Bool True Y. Bool + Bool - Bool Digit T - Folse	T-APP