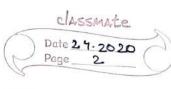


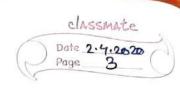
| TYPE SAFETY OF 1F+7 |
|----------------------------------------------|
| |
| - semantic domains of values. |
| - syntax of IF+/ |
| rewrite rules. |
| reduction |
| - normal forms VAL, DIV/O ATM. |
| partition theorem for normal forms. |
| - preservation theorm for VAL*, DIV/0*, ATM* |
| - DESTINY thm. |
| twell typing |
| type preservation thm & type safety |
| type progress thm |
| |
| |
| TYPE SAFETY |
| |
| semantic domains. |
| · · · · · · · · · · · · · · · · · · · |
| n & NOM |
| b & BOOL = & true, false} |
| v e VAL |
| |
| v::= n b |
| |

+: [MUN, MUN] -+ NUM

MON - [MON MON]: 1



| | SYNTAX OF 1F+1 |
|---|-------------------------------------------------------------------------------------------|
| | |
| | e:= ñ e ExP |
| | 6 |
| | e (+) e |
| | e/)e |
| | (if) e e e |
| | |
| | |
| | |
| | LITERAL EXPRESSIONS |
| | |
| | - VAL e VAL |
| | V VAL |
| | L e VAL |
| | VAL |
| | |
| | |
| | |
| | SEMANTICS |
| | |
| | rewrite roles e cye' |
| | |
| _ | $n_1 + n_2 \longrightarrow n_1 + n_2$ PLUS |
| | |
| | $\vec{n}_1/\vec{n}_2 \longrightarrow \vec{n}_1/\vec{n}_2$ DIV provided $\vec{n}_2 \neq 0$ |
| _ | |
| _ | if true e2 e3 C> e2 IF-TRUE |
| _ | |
| | if balse e2 e3 L7 e3 IF-FALSE |



SEMANTICS

reduction system

e -> e1

e -> e' REW

PLUS-LEFT

e, -> e,'
e,+e2 -> e,+e2

 $e_1 VAL$ $e_2 \rightarrow e_2'$ PLUS-RIGHT $e_1 + e_2 \rightarrow e_1 + e_2'$

notes reduction is deterministics
left operand reduced first

 $e_1 \rightarrow e_1'$ DIV-LEFT $e_1 \bigcirc e_2 \rightarrow e_1' \bigcirc e_2$

e, De2 -> e, De2 DIV-RIGHT

 $e_1 \rightarrow e_1'$ $(if) e_1 e_2 e_3 \rightarrow (if) e_1' e_2 e_3$ $(if) e_1 e_2 e_3 \rightarrow (if) e_1' e_2 e_3$

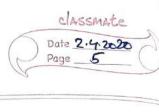


DIV BY ZERO ERROR

| | o ope&(+), (1) |
|-----------------------------------------|---------------------------------------|
| n/0 DIV/0 | |
| | |
| 2 7 7 | Diane |
| e1 VAL e2 DIV/D | RIGHT |
| e1 OP e2 DIV/0 | |
| * * * * * * * * * * * * * * * * * * * * | 2. 25 a |
| e, Divlo | LEFT |
| e, Op e2 DIV/0 | |
| G DIV/O | |
| (b) e, e2 e3 DIV/0 | |
| | · · · · · · · · · · · · · · · · · · · |
| e DIV/O | P DIV/0 |
| DIV | 6 DIN/O |
| res se a S | 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 |
| stock atl | |
| STOCK OU | |

e DIV/O means a e />

2 reason is DIV/o



ex if (4 + 3/0) /2 7+8 false DIV/0

(0) D 3/0 DIV/0 (RIGHT) (4 + 3/0) DIV/0

3 (4+310)/2 DIV/O (LEFT)

(5) if (4+310)12 7+8 palse DIV/O (IF)

ex a DIV/o can occor multiple times.

0 4/0 DVIV/0 (0)

4/0 + 3/0 DW/0

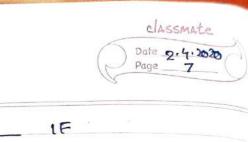
@ 4/0 + 3/0 DIVIO (LEFT)

- balse + 310 @ 3/0 DIV(0 (0)

false VAL VAL: VAL ?? false + 3/0 DIV/O (RIGHT)

| e' DIV/0 e *> e' | E DIV/O* |
|-------------------------|----------------------|
| e DIV/0* | <u> </u> |
| | |
| DIV/0* | |
| DIV/0* | |
| | |
| e simplifies to an expr | ession stuck due |
| to DIV/o error, | |
| | |
| * | T V |
| e ATM ATM | Q ATM |
| ATM | One hive |
| * | arg type mismatch |
| Be | DOL-NOM |
| bi OP no ATM | |
| | |
| No | M-BOOL |
| FI OP BY ATM | |
| | |
| | 00L-BOOL |
| 6, 6P by ATM | |
| | |
| - | |
| | FT |
| e1 OP e2 ATM | |
| O. ATAM | _ |
| e2 ATM RI | GHT |

ATM



(ib) n, e2 e3 ATM

intuitively, e is stuck due to argument type

mis moter,

e' ATM e *> e' *

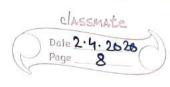
simplification of e results in an expression

that is stuck due to an ARCT TYPE MISMATCH

ex +rue/0 + 3/0

1) true/ O ATM (BOOL-NUM) 1 true(0 + 3/0 ATM (LEFT)

.. ATM truelo + 3/0 ATM



| | normal form | description | | |
|---|----------------|--------------------------------|--|--|
| | VAL | values | | |
| * | DW/o | Stuck at DIV/o | | |
| | ATM | stuck due to arg type mismatch | | |
| | | | | |
| | e val => | e A | | |
| | e 011/0 => | e +> | | |
| | e atm => | e / > | | |
| | | | | |
| | SPANNING LEMMA | | | |
| | is e →, | De VAL, or | | |
| | | 2) e DIV/0, or | | |
| | | De ATM | | |
| _ | | | | |
| | + | | | |