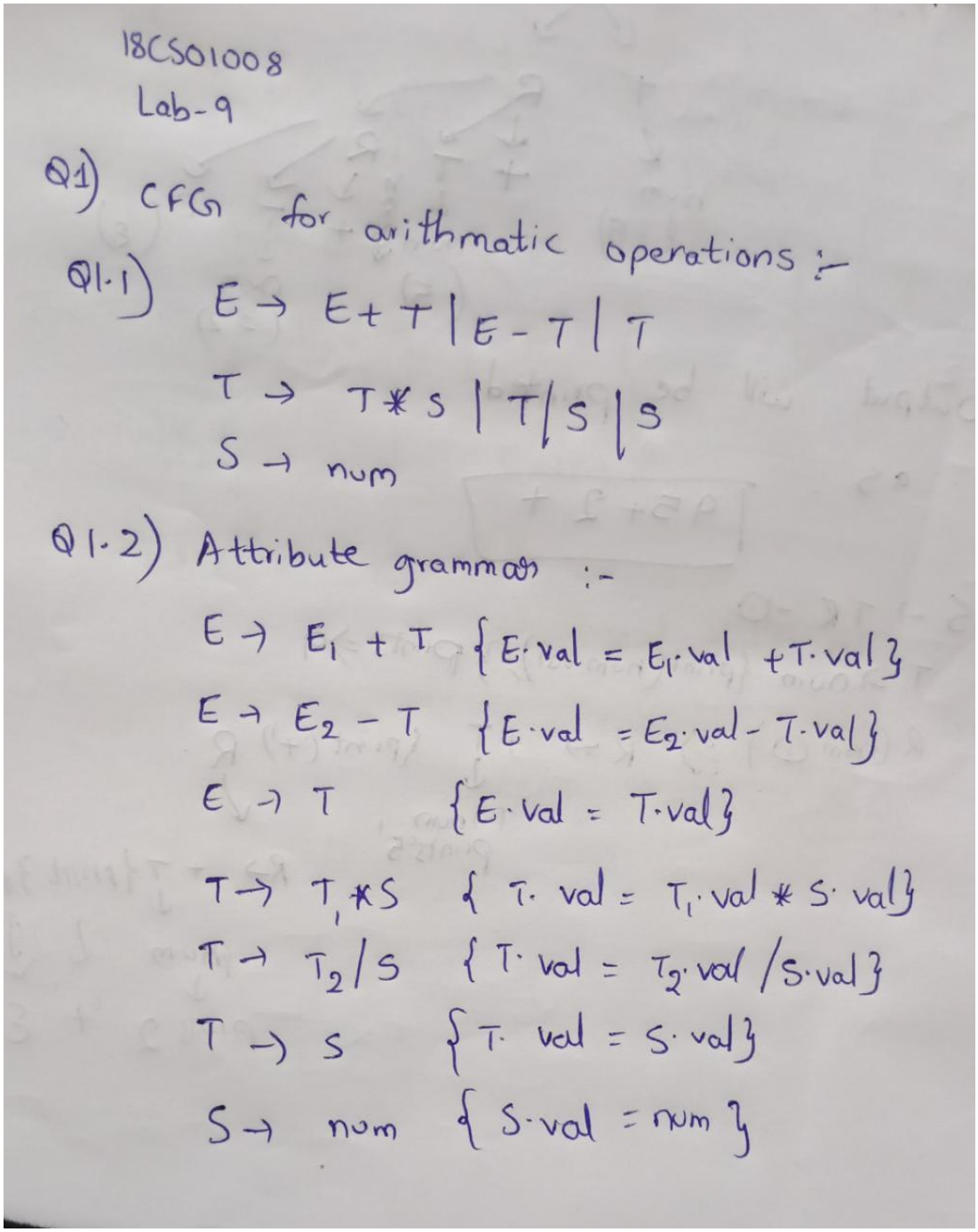
Compiler Design Lab
Lab-9
Attribute Grammars

Name: T.V.S.S.Sripad
Roll Number: 18CS01008

Q1)



18CS01008
Lab-9

Q1) CFG for arithmatic operations :-

Q1.1) $E \rightarrow E + T \mid E - T \mid T$

$T \rightarrow T * S \mid T / S \mid S$

$S \rightarrow num$

Q1.2) Attribute grammar :-

$E \rightarrow E_1 + T \quad \{E.val = E_1.val + T.val\}$

$E \rightarrow E_2 - T \quad \{E.val = E_2.val - T.val\}$

$E \rightarrow T \quad \{E.val = T.val\}$

$T \rightarrow T_1 * S \quad \{T.val = T_1.val * S.val\}$

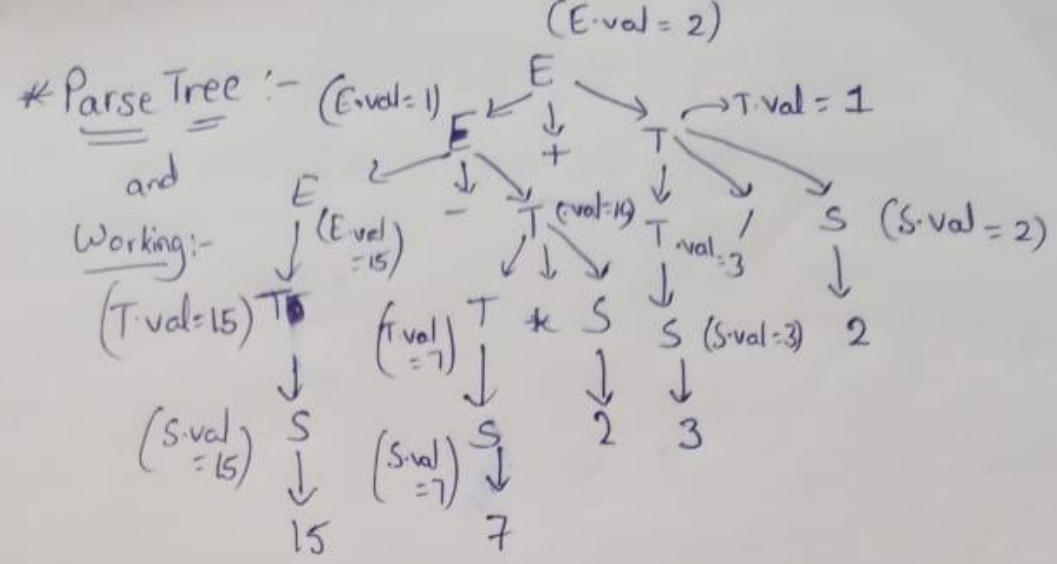
$T \rightarrow T_2 / S \quad \{T.val = T_2.val / S.val\}$

$T \rightarrow S \quad \{T.val = S.val\}$

$S \rightarrow num \quad \{S.val = num\}$

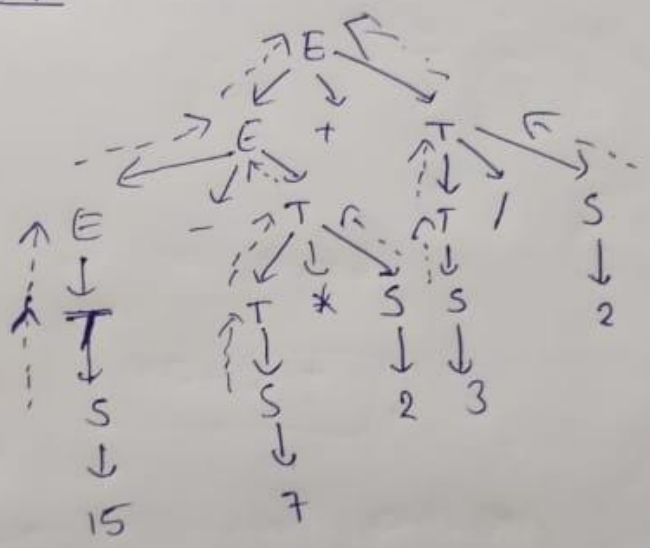
Example :- 15- 7 * 2 + 3/2

* Parse Tree :- and Working :-



Dependency graph :-

In this graph, dotted lines represent



NOTE:

1) Negative Integers are also supported in the attached implementation
2) Along with the standard mathematical operations like addition, subtraction, multiplication, division, **Mod operation, Power operator,** Brackets are also considered in the attached implementation.
3) Mod operator is to be written as : 5 % 2
4) Power is to be written as : 2 **3 (which is 2 raised to power of 3)
5) The final attribute grammar is:-

E -> E '+' T { E.val = E1.val + T.val }
| E '-' T { E.val = E2.val - T.val }
| T { E.val = T.val}

T -> T '*' S { T.val = T.val * S.val }
| T '**' S { T.val = power(T.val , S.val) }
| T '%' S { T.val = T.val % S.val }
| T '/' S { T.val = T.val / S.val }
| S {T.val = S.val }

S -> num { S.val = num }
| '-' num { S.val = -num }
| '(' E ')' { S.val = E.val }

Steps to run:-

Run the run.sh script using the command ./run.sh.

Screenshots of Sample Executions:

1) Input: 1+(2*8/4)+(3**2)

2) Input: 1-9/3+2*3+7%3