GROUP-50

111

COMPILER CONSTRUCTION

GRAMMAR (FIRST & FOLLOW SETS)

-> TEAM MEMBERS

- 1) Harshit Gupta (201982A70898P)

- 2) Housh Pandey (2019B3A70489P) 3) Asyan Puwan (2019B3A70470P) 4) Tejas beshbande (2019B3A70562P)
 - 5) KNISH VOTA (2019B3A70819P)

MODIFIED GRAMMAR modified or added.

() - Means that the

< program > -> < module Declarations > Cother Modules > Cobio en Module > Cother Modules >

- · <module beclarations > -> <module beclaration> <module beclarations > 1 &
- conducted claration> > DECLARE MODULE ID SEMICOL
- CotherHodules>→ Cmodule>CotherHodules>1E
- 5. < doiOUMobile> -> DRIVERBEF DRIVER PROGRAM DRIVERENDOEF CWOODLEDEJ?
- 6. ¿module > → DEF MODULE ID ENDBEF TAKES INPUT SOBO cinbul-plus > SOBE SEMICOL cret> cmodule bej>
- 7. CART >> RETURNS SOBO CONTROL PLIAT > SOBC SEMICOL IE
- 8. < input-plint> -> ID COLON < dataType> < lefteel>
- 9 (leftrect> -> comma IN colon cootestype> cleftrect>1 €
- 101 content-plint> > 10 colon ctype> cleft Rec 2>
- Clytrecz> -> comma ID colon Ctype> Clytrecz> 12
- 12 colatatype> -> ARRAY SOBO commange> SOBC OF Ctype>
- 13. Cornhange> -> cinden > RANGEOP (inden >
- 14. Ctype> -> INTEGER | REAL | BOOLEAN
- 15. cmodulebef> → START Catatements > END
- 16. Catatements> → Catatement> Catatements>1E
- 17. Cotatement > -> Ciostmt > 1 Snimple Stut > 1 < declare Stut > 1 < conditional Stut > 1 /iterativestmt>
- 18. Ciostmt> -> GET_VALUE BO ID BL SEMICOL PRINT BO COON> BC SEMICOL
- 161 Cvan > -> cvar_num > 1 c boolean Court>
- MUCHALMONS STE CONTRACTOR
- 201 c boolean Court > -> TRUE | FALSE
- 22. cwhich Id > -> SOBO cinden > SOBC | E
- 23. Crimplestut >- xonignmentstat > 1 (module Recurstant >
- 24. canignmentstat> -> ZD (whichstat)
- 26. Cwhich Strut >> walue IDStrut > 1 < Ivalue ARR Strut >
- 26. ChalmeIDStut> -> ACSIGNOP Confirmion > SEMICOL

```
Loolne ARRStad > -> SOBO CINDEN > SOBC ASSIGNOP CENTRATION > SEMILOR

Cinden > -> NUM | IB

Conduct Remark > -> Coptional > USE MODELE ID WITH PARAMETERS

CidList > SEMILOR

SO Coptional > -> SOBO CIDLIST > SOBC ASSIGNOP | E

SI CidList > -> IB Cleftec 3 >

SY Cleftec 3 > -> COMMA IB Cleftec 3 > | E

SY Comparison > -> ConthBood Endor > | Cumary Endor >

CUmary Endor > -> Cumary OP > Ctompalead >

Comparison > -> ConthBood Endor > BC | Cumary Cumary OP > -> PLUS | MINUS
```

37. Cavith Boot Embor > -> Chome Term > Cleft Rec 4>

38. (leftlec 4) -> (logical 0) > Chometerm > (leftlec 4) | E 39. (Loome Term > -> Cavithmetictuper > Cleftlec 7 > 1 (booleanlaunt > Cleftlec 3)

40. (left Rec 3) -> crelational 0>> Canithmetic Eupr> (left Rec 3>1 &

Mr. Covillametic Empr> -> Ctorm> Cleftrecb>

42. Cleftrec6> -> cop-lawer> (term> (leftrec6> 1E

43. ctorm> -> Claston> Clytred>>

44. Cleftec 7> -> Cob-higher > Clastor > Cleftec 7>1E

45. cyclor> -> BO cowthmetic Entr> BC / Cournum>

Mr. COLLEW > -> PLUS IMINUS

42. Cobfleigh > -> MULIDIU

48. clogicalOp> -> ANDIOR

49. L'relational Op> -> LT | LE | GT | GE | EQ | NE

50. cdeclarestmt> -> DECLARE CIOLIST> COLON COLATATYDE> SEMICOL

51. Cconditionalstant) -> SWITCH BO ID BC START (casestant) code gaults)

52. cease Stut > -> CASE coalue > COLON & statements > BREAK

SEMICOL Cleft Rec 8>

SREAK SEMICOL Cleft Rec 8>

SREAK SEMICOL Cleft Rec 8>

May 2

May 2

colue> -> NOM | TRUE | FALSE

colon catalements> BREAK SEMICOL | E

colon catalements> BREAK SEMICOL | E

colon catalements> BC START catalements>

colon control BC START catalements>

END | WHILE BO control Control BC START catalements>

END | WHILE BO control Control BC START catalements>

END

67. Crange > -> NUM RANGEOP NUM

(-) denotes rules which were modified or added

	34	五、5公主,1944年4月		
		Non Terminals	FIRST	FOLLOW
	Alex-	program	1 DECLARE 3	{4 b
		module Declarations	f DECLARE &	{DEF, DRIVERDEF, \$}
		module Declaration	{DECLARE}	(DECLARE DEFT
			(50000000	NOTUEP DET ? *)
1	V	other modules	{DEF, E}	(DRIVERDEF. \$4
1	5. 0	driver Module	(DRIVERDEF)	EDEF & }
	6.	module	1 DEF3	{DEF · \$ }
	7.	ret	{ RETURNS }	ISTART Y
	8.	left Rec 1	{comma }	150BC7 {50BC}
ŝ	9.	output-plist	{ID}	(sabc)
	10.	leftRec 2	f commaney	(50864
3	W	data Type	{ ARRAY }	(SUBC)
	12.	input-plist	& NUM , ID3	(sabc)
	13.	arr-range	{ INTEGER , REAL ,	(sanc)
	14.	type	BOOLEAN	
	15.	module Def	START Y	(DEF- \$4
	16.	stadements	LOST-VALUE, PRINT, ID,	BREAK END)
		SWEITENS	SONO DECLARES SWITCH	2
		W.7	Eno WHILE TE	PRINTAIDS
	17.	statement	SCHET-VALUE, PRINT, ID	1 COAD - DECLARES SWITTEN
			5080, DECLARE, SWITCE FOR, WHILE &	T-A LOHTLE &
		A COMPANY OF THE PARTY OF THE P	{GET-VALUE , PRINT }	16 TT VALUE OPETIVI 9 TUT
	18.	iostmt	SCHELLANGOE ALKEN 1	SOBO DECLARES SULLE
				FOR , WHILE ?
			(ID. NUM, RNUM, TEL	UE, EBC3
	19.	Vari	FALSE?	{BC, SEMICOLOND}
	. 5		{ID= NUM , RNUM }	
	20.	var-num	{TRUE & FALSE }	(BC)
- 2	21.		15000 9 & y	(BL, SEMILOLOAD)
3	22		The state of the s	(BREAK TENDS
	23	. simplestmt	(ID, SOBO, USES	(BREAK END)
	24	Ar L. A	(ID)	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
		A A	{ASSIGNOP, SQBO}	The state of the s
	25	. which serve	{ASSIGNOP}	(BREAK TEND)
	26	. Ivalue IDstat	{ 50 Bo}	(BREAK & END)
	22	. Ivalue ARRSTAND	& NUM, IDY	ESQBE RANGEOR?
3	25	The state of the s	(100.000)	{BREAK NEND}
		Jule Perus Strat	¿ sabon use}	Plage 4
	20.	1. Moonicionse	(*)	

	Non Terminals	FIRST	FOLLOW
1	optional idList	(SOBO, E)	{USE} {SEMILOL, SOBCO
	leftlec3 expression	ECOMMA = E } EPLUS OMINUSO BOO IDO NUMO RNUMO	(SEMICOLY SOBCYCOLON)
35. 36.	unary Expr term Ahead unary Op arith BoolExpr	TRUE, FALSE? { PLUS, MINUS; { BO, ID, NUM, ENUM; { PLUS, MINUS; { BO, ID, NUM, ENUM;	[SEMILOLY
38. 39.	leftRec 4 some Term	TRUE & FALSE & {AND, OR & E Y {BO, ID, NUMBERUM	{SEMICOLOBC} {SEMICOLOBC}
	left rec 5	TRUE, FALSE & {LT, LE, GT, GE, EQ, N {BO, ID, NUM, RNUM}	(BC) (SEMICOL & BC) (BC) SEMICOL & LTO LE, GTOGEO NOONE)
42. 43.	tesm left Rec 6	{ BO, ID, NUM, ANUM?	PLUS, MINUS, BC, SEMICOL, LT, LE, GT, GE, NO, NE)
hy.	left lec 7	{ MUL DIVAE }	EPLUS a MIUS a BC a SEMILOL a LT a LE , GT,
W5.	factor	{BO, ID, NUM, RNUM}	GEONQONE } {MUL, DIVO PLUS, MING, BCOSEMICOLOLITALE, GTOGEONONOS
46.	oplow	{PLUS am INUS }	(BO, ID, NUM-RNUM)
47.	optigh	{ MUL, DIV }	(BU, ID, NUMARNUM
48.	logical Op	¿ AND ORY	(BO, ID. NUM, RNUM, TRUE, FALSE)
49.	relationalOp	{ LTALEAGTAGE, NO,	NES {BOITO, NUM, RNUM?
50.	declare Strnt	{DECLARE}	(GET-VALUE, PRINT, ID, SOBO, DECLARE, SWITCH FOR, WHILE) MASS

Non-Terminals	FIRST	FOLLOW
conditional Stret	ESWITCH }	GET-VALUE, PRINT, ID, SOBO, DECLARE, SWITCH
case Stmt	{CASE }	FOR, WHILE J
left Rec8	& CASE , E }	{DEFAULT, AND }
value	I NUM, TRUE, FALLE	(capy }
default	EDEFAULT, EZ	END }
iterativeStmt	(FOR WHILE)	(GET-VALUE, PRINT,
range	{ NUM }	ID, SQBO, DECLARE, SWITCH, FOR, WHILE {BC}