

Workshop 2 → Computer Science 3.

Laura Daniela Muñoz.

1. For each one of the next cases define a regular expression as used in a compiler based on the Python re library.

i) Identifier → A regular expression to match valid identifiers.

Identifier = `r"^[a-zA-Z_][a-zA-Z0-9_]*$"`

ii) Integer Literal → Match Integer literals.

Integer_literal = `r"^-?\d+$"`

iii) Floating point Literal → Match Floating points

Floating_point_literal = `r"^-?\d+\.\d+$"`

iv) String Literal → String literals enclosed in double quotes.

string_literal = `r"^\".*\"$"`

v) Single line → Single line comments, ~~enclosed in~~ starting with '///'.

single_line_comment = `r"^///.*"`

vi) Multi_line comment → Multiline comment enclosed in '/* */'

multi_line_comment = `r"/*[\s\S]*?*/"`

vii) WhiteSpace → (space, tabs, newlines).

whiteSpace = `r"\s+"`

viii) Operators → Common operators.

Operator = `r"^(==|!=|<=|>=|\\|\\+|\\-|*|\\/|<|>|=)$"`

///

1x) Keywords \rightarrow match keyword ('if', 'else', 'while', 'return')

Keyword = $r''^{\wedge}(if|else|while|return)\$''$

x) Hexadecimal Literal

Hexa_literal = $r''^{\wedge}0[xX][0-9a-fA-F]^+\$''$

2. Be G a context-free grammar with the following productions =

$S \rightarrow \text{Program}$

$\text{Program} \rightarrow \text{StatementList}$

$\text{StatementList} \rightarrow \text{Statement} \mid \text{StatementList } \langle \text{lambda} \rangle$

* $\text{Statement} \rightarrow \text{Assignment} \mid \text{IF statement} \mid \text{While statement} \mid \text{Return Statement}$

$\text{Assignment} \rightarrow \text{Identifier } "=" \text{ Expression } ";"$

$\text{IF statement} \rightarrow "if" "(" \text{ Expression } ")" "{" \text{ StatementList } "}"$
Else Part.

$\text{Else Part} \rightarrow "else" "(" \text{ StatementList } ")" \mid \langle \text{lambda} \rangle$

$\text{While statement} \rightarrow "while" "(" \text{ Expression } ")" "{" \text{ StatementList } "}"$

$\text{Return Statement} \rightarrow "return" \text{ Expression } ";"$

$\text{Expression} \rightarrow \text{Term} \mid \text{Expression } "+" \text{ Term} \mid \text{Expression } "-" \text{ Term} \mid \langle \text{lambda} \rangle$

$\text{Expression}' \rightarrow "+" \text{ Term } \text{Expression}' \mid "-" \text{ Term } \text{Expression}' \mid \langle \text{lambda} \rangle$

$\text{Term} \rightarrow \text{Factor} \mid \text{Term}'$

$\text{Factor Term}' \rightarrow "*" \text{ Factor } \text{Term}' \mid "/" \text{ Factor } \text{Term}' \mid \langle \text{lambda} \rangle$

$\text{Factor} \rightarrow "(" \text{ Expression } ")" \mid \text{Identifier} \mid \text{Number}$

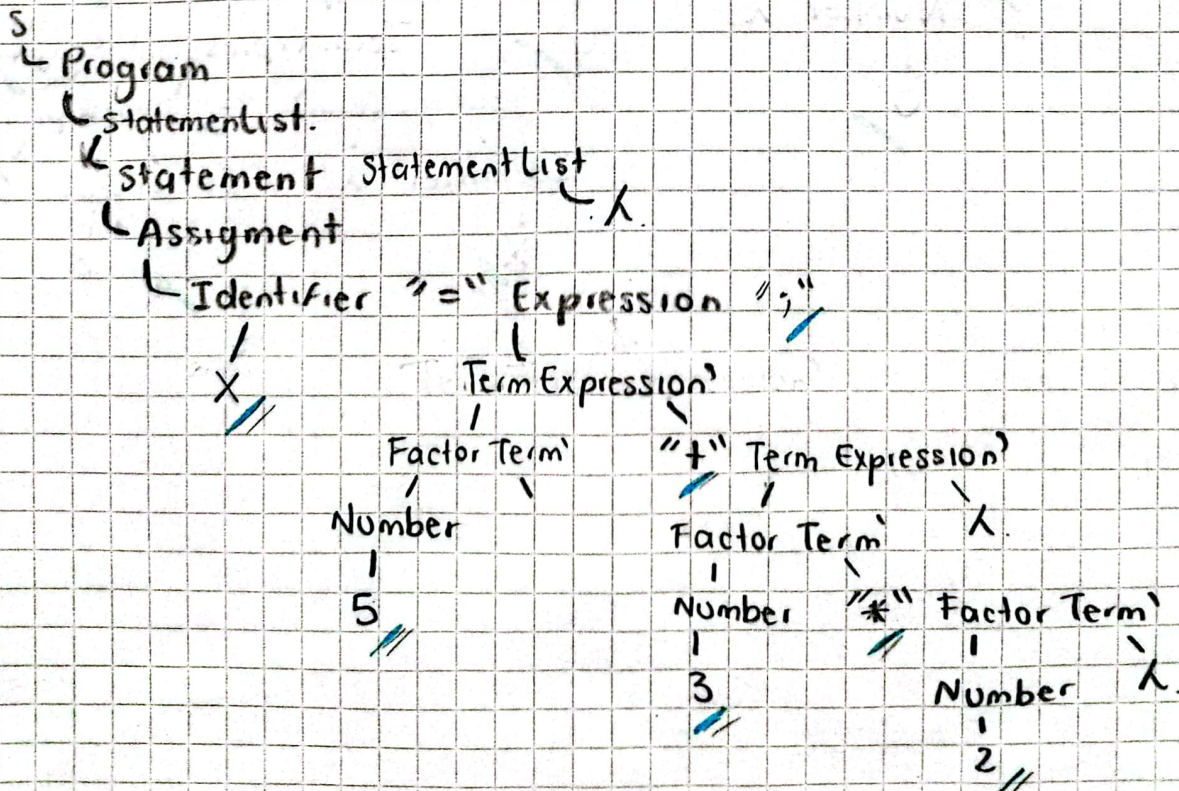
$\text{Identifier} \rightarrow [a-zA-Z_][a-zA-Z0-9_]*$

$\text{Number} \rightarrow [0-9]^+$

→ Create derivation trees for the following statements =

a) Exercise 1 =

$X = 5 + 3 * 2;$




```

1 if (x > 0) {
    y = x - 1;
} else {
    y = 0;
}

```

→ There's not logical operator in the context free grammar.

$$\text{Expression} = \text{"+" Term Expression} \mid \text{"-" Term Expression} \mid \text{logical Op Term Expression} \mid \wedge.$$

Logical Operator = ">" | "<" | "≥" | "≤" |
"!=" | "λ."

S' Program

Statement List

Statement Statementlist.

IF statement

"if" "(" Expression ")" "{" Statement List "}" ElsePart

Term Expression?

Statement	Statement list
1. $x \leftarrow 1$	1. $x \leftarrow 1$
2. $y \leftarrow 2$	2. $y \leftarrow 2$
3. $x \leftarrow x + y$	3. $x \leftarrow x + y$
4. $y \leftarrow y + x$	4. $y \leftarrow y + x$
5. $x \leftarrow x - y$	5. $x \leftarrow x - y$
6. $y \leftarrow y - x$	6. $y \leftarrow y - x$
7. $x \leftarrow x * y$	7. $x \leftarrow x * y$
8. $y \leftarrow y * x$	8. $y \leftarrow y * x$
9. $x \leftarrow x / y$	9. $x \leftarrow x / y$
10. $y \leftarrow y / x$	10. $y \leftarrow y / x$
11. $x \leftarrow x \% y$	11. $x \leftarrow x \% y$
12. $y \leftarrow y \% x$	12. $y \leftarrow y \% x$
13. $x \leftarrow x \& y$	13. $x \leftarrow x \& y$
14. $y \leftarrow y \& x$	14. $y \leftarrow y \& x$
15. $x \leftarrow x y$	15. $x \leftarrow x y$
16. $y \leftarrow y x$	16. $y \leftarrow y x$
17. $x \leftarrow x \oplus y$	17. $x \leftarrow x \oplus y$
18. $y \leftarrow y \oplus x$	18. $y \leftarrow y \oplus x$
19. $x \leftarrow x \> y$	19. $x \leftarrow x \> y$
20. $y \leftarrow y \> x$	20. $y \leftarrow y \> x$
21. $x \leftarrow x \< y$	21. $x \leftarrow x \< y$
22. $y \leftarrow y \< x$	22. $y \leftarrow y \< x$
23. $x \leftarrow x \geq y$	23. $x \leftarrow x \geq y$
24. $y \leftarrow y \geq x$	24. $y \leftarrow y \geq x$
25. $x \leftarrow x \leq y$	25. $x \leftarrow x \leq y$
26. $y \leftarrow y \leq x$	26. $y \leftarrow y \leq x$
27. $x \leftarrow x == y$	27. $x \leftarrow x == y$
28. $y \leftarrow y == x$	28. $y \leftarrow y == x$
29. $x \leftarrow x \neq y$	29. $x \leftarrow x \neq y$
30. $y \leftarrow y \neq x$	30. $y \leftarrow y \neq x$
31. $x \leftarrow x \> y \& y \> x$	31. $x \leftarrow x \> y \& y \> x$
32. $y \leftarrow y \> x \& x \> y$	32. $y \leftarrow y \> x \& x \> y$
33. $x \leftarrow x \< y \& y \< x$	33. $x \leftarrow x \< y \& y \< x$
34. $y \leftarrow y \< x \& x \< y$	34. $y \leftarrow y \< x \& x \< y$
35. $x \leftarrow x \geq y \& y \geq x$	35. $x \leftarrow x \geq y \& y \geq x$
36. $y \leftarrow y \geq x \& x \geq y$	36. $y \leftarrow y \geq x \& x \geq y$
37. $x \leftarrow x \leq y \& y \leq x$	37. $x \leftarrow x \leq y \& y \leq x$
38. $y \leftarrow y \leq x \& x \leq y$	38. $y \leftarrow y \leq x \& x \leq y$
39. $x \leftarrow x == y \& y == x$	39. $x \leftarrow x == y \& y == x$
40. $y \leftarrow y == x \& x == y$	40. $y \leftarrow y == x \& x == y$
41. $x \leftarrow x \neq y \& y \neq x$	41. $x \leftarrow x \neq y \& y \neq x$
42. $y \leftarrow y \neq x \& x \neq y$	42. $y \leftarrow y \neq x \& x \neq y$
43. $x \leftarrow x \> y \& y \> x \& x \< y \& y \< x$	43. $x \leftarrow x \> y \& y \> x \& x \< y \& y \< x$
44. $y \leftarrow y \> x \& x \> y \& y \< x \& x \< y$	44. $y \leftarrow y \> x \& x \> y \& y \< x \& x \< y$
45. $x \leftarrow x \geq y \& y \geq x \& x \leq y \& y \leq x$	45. $x \leftarrow x \geq y \& y \geq x \& x \leq y \& y \leq x$
46. $y \leftarrow y \geq x \& x \geq y \& y \leq x \& x \leq y$	46. $y \leftarrow y \geq x \& x \geq y \& y \leq x \& x \leq y$
47. $x \leftarrow x == y \& y == x \& x \neq y \& y \neq x$	47. $x \leftarrow x == y \& y == x \& x \neq y \& y \neq x$
48. $y \leftarrow y == x \& x == y \& y \neq x \& x \neq y$	48. $y \leftarrow y == x \& x == y \& y \neq x \& x \neq y$
49. $x \leftarrow x \> y \& y \> x \& x \< y \& y \< x \& x \geq y \& y \geq x$	49. $x \leftarrow x \> y \& y \> x \& x \< y \& y \< x \& x \geq y \& y \geq x$
50. $y \leftarrow y \> x \& x \> y \& y \< x \& x \< y \& y \geq x \& x \geq y$	50. $y \leftarrow y \> x \& x \> y \& y \< x \& x \< y \& y \geq x \& x \geq y$
51. $x \leftarrow x \geq y \& y \geq x \& x \leq y \& y \leq x \& x == y \& y == x$	51. $x \leftarrow x \geq y \& y \geq x \& x \leq y \& y \leq x \& x == y \& y == x$
52. $y \leftarrow y \geq x \& x \geq y \& y \leq x \& x \leq y \& y == x \& x == y$	52. $y \leftarrow y \geq x \& x \geq y \& y \leq x \& x \leq y \& y == x \& x == y$
53. $x \leftarrow x \neq y \& y \neq x \& x \> y \& y \> x \& x \< y \& y \< x$	53. $x \leftarrow x \neq y \& y \neq x \& x \> y \& y \> x \& x \< y \& y \< x$
54. $y \leftarrow y \neq x \& x \neq y \& y \> x \& x \> y \& y \< x \& x \< y$	54. $y \leftarrow y \neq x \& x \neq y \& y \> x \& x \> y \& y \< x \& x \< y$
55. $x \leftarrow x \> y \& y \> x \& x \< y \& y \< x \& x \geq y \& y \geq x \& x == y \& y == x$	55. $x \leftarrow x \> y \& y \> x \& x \< y \& y \< x \& x \geq y \& y \geq x \& x == y \& y == x$
56. $y \leftarrow y \> x \& x \> y \& y \< x \& x \< y \& y \geq x \& x \geq y \& y == x \& x == y$	56. $y \leftarrow y \> x$

else = {"statement list"}

Statement Statement by
Assignment

Assignment 10 A.
Identify "5" From "all"

Identifier "=" Expr "++"

Term Expr

Term Expr
factor 'A

Factor X
Number
L.O.

CO
Expressions

Expression

2. 1.

[illegible]

人

[illegible]

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

[illegible][illegible][illegible][illegible]

mScanner

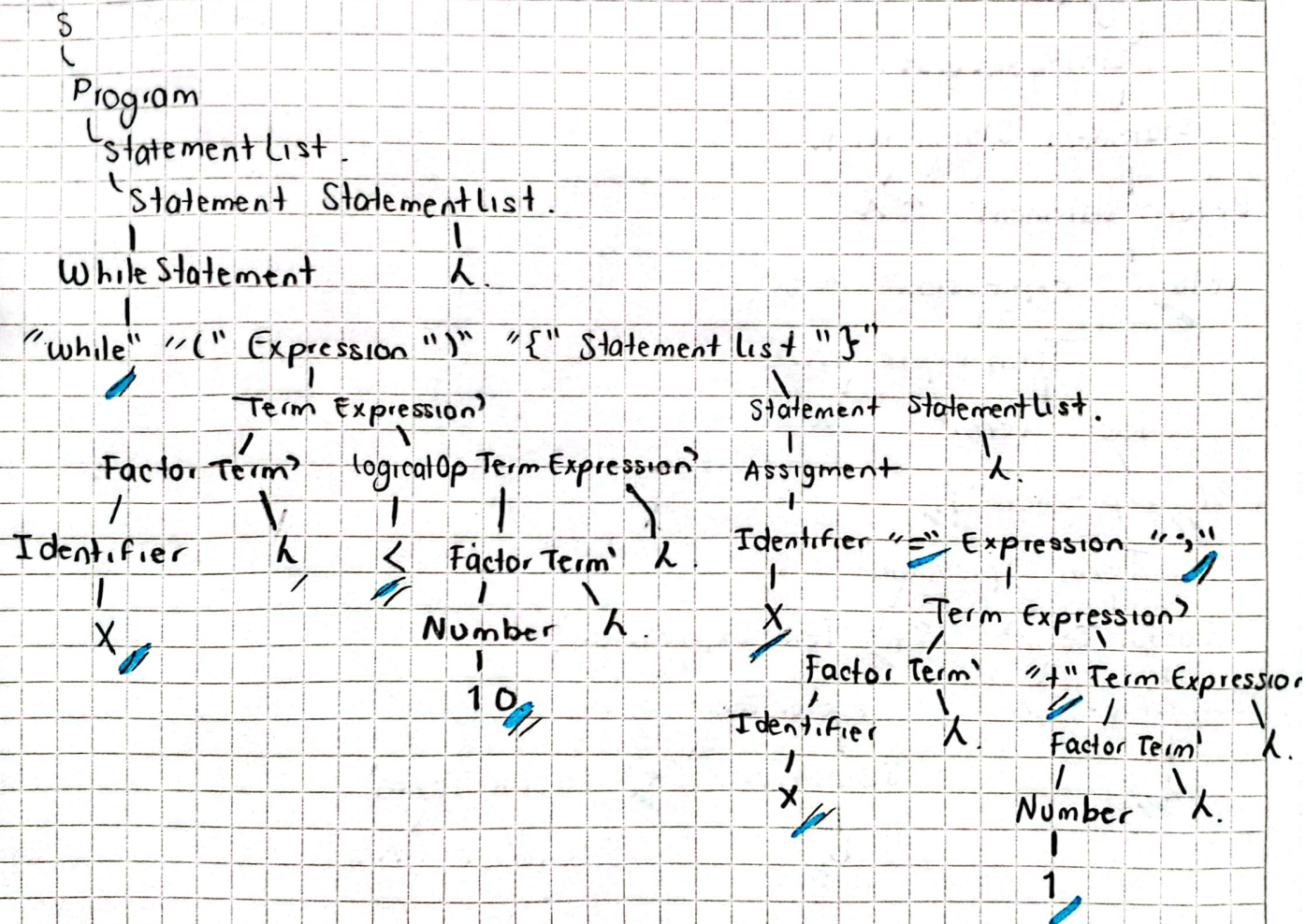
Scanner

c). Exercise 3.

```

while (x < 10) {
    x = x + 1;
}

```



d). Exercise 4.

return (a+b) * c ;

