Chapter 20: System software: Answers to Worksheet questions

1 **a** NewValue

←

SQRT

(

OldValue

)

+

SQRT

(

OldValue

)

*

PercentIncrease

/

100

Fifteen tokens in all.

b
- Only identifiers are entered into the symbol table. This means:
- NewValue, SQRT, OldValue and PercentIncrease.
- Included in the table for each identifier is relevant data such as data type, where declared if the identifier is a variable, and where first assigned a value.

2 **a** During the back-end synthesis stage.

b One type of optimisation relates to inefficiencies in the original source code. A typical example is incorrect positioning of assignment statements when a loop structure is used. Another is where a sequence of assignment statements could be changed because there is some common expression that could be defined and used several times.

The other type of optimisation concerns the machine code generated, which might not make optimum use of the registers in the processor or might make more calls to memory than are really necessary.

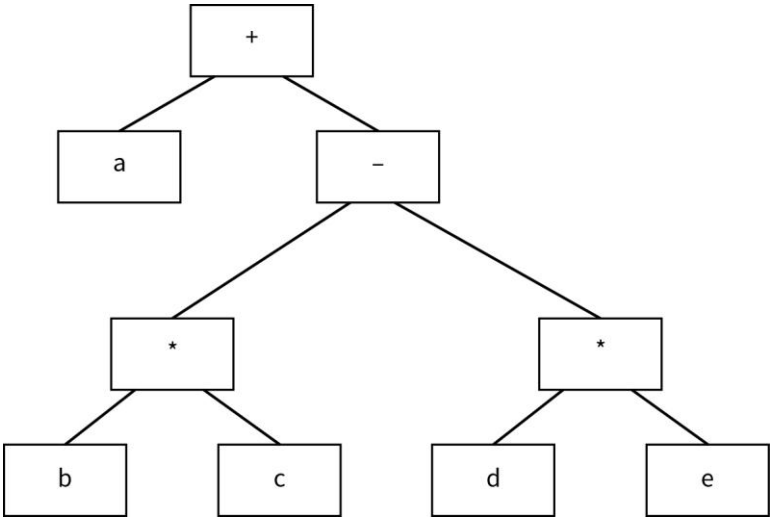
c An interpreter reads a line of source code, and if no syntax error is detected it converts the line of source code to intermediate code and then executes it. Therefore the interpreter cannot make judgements about sections of code possibly needing reorganisation.

3 2, 7, 8, 4, 9, 3, 5, 6, 1, 10

Learners need to know some specialised names and understand their meanings. The order of the four stages makes sense once this understanding is reached. It is important to remember that syntax analysis and parsing

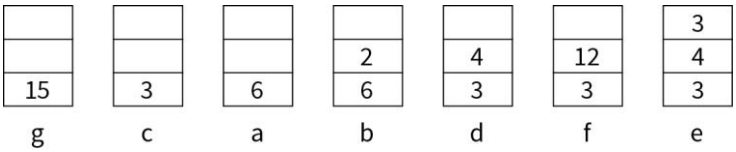
mean the same thing. Overall the front-stage analysis aims at taking the source code, documenting it and converting it to intermediate code. The back end synthesis stage can then use the intermediate code and the documentation to create object code.

4



Construction of the tree must start with the first lowest precedence operator at the root. This must lead to any other lowest precedence operators before the higher precedence operators are entered into the tree. The symbols representing variables are all positioned as leaf nodes. As a useful exercise the tree can now be used to create a RPN version of the expression. The RPN form of the expression can be extracted by a post-order traversal. This starts at the lowest leaf to the left of the root and then uses left–right–'root' ordering successively. Note 'root' is in quotes because strictly there is only one root, below this there are what might be called sub-roots.

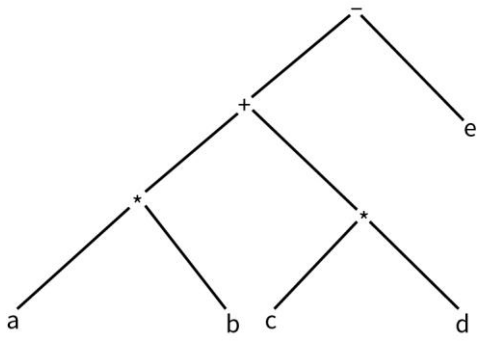
5



6 1, 9, 2, 6, 3, 10, 8, 4, 7, 5, 11, 13, 12

This question requires following the rules. These rules involve decisions about operator precedence in an infix expression. Here * is higher precedence than + or -.

7 a

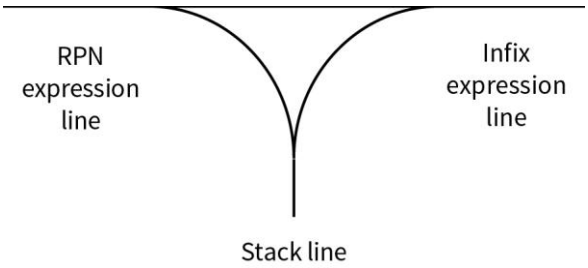


The tree above is arranged with the sensible choice of the low precedence operator as root.

Post-order traversal produces the RPN as:

ab*cd*+e-

b



Learners could answer the question by annotating this diagram.

The alternative is to use a table:

Infix line	Stack line	RPN line
a * b + c * d - e		
* b + c * d - e		a
b + c * d - e	*	a
+ c * d - e	*	ab
c * d - e	+	ab*
* d - e	+	ab*c
d - e	+*	ab*c
- e	+*	ab*cd
e	-	ab*cd*+
	-	ab*cd*+e
		ab*cd*+e-

c 6 + 20–6 = 20

d