AQA Computer Science A-Level

4.4.3 Context-free languages

Past Paper Questions

June 2012 Comp 3

2 **Figure 1** shows some production rules that have been used to define the syntax of valid mathematical expressions in a particular programming language.

Figure 1

```
<expression> ::= <factor> | <factor> * <factor> | <factor> / <factor>
<factor> ::= <term> | <term> + <term> | <term> - <term>
<term> ::= - <expression> | <number>
<number> ::= <digit> | <digit> <number>
<digit> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
```

2 (a) What notation method has been used in **Figure 1**?

.....

(1 mark)

2 (b) Complete **Table 1** by writing **Yes** or **No** in the empty column to indicate whether or not the strings are valid examples of the statement types from **Figure 1**.

Table 1

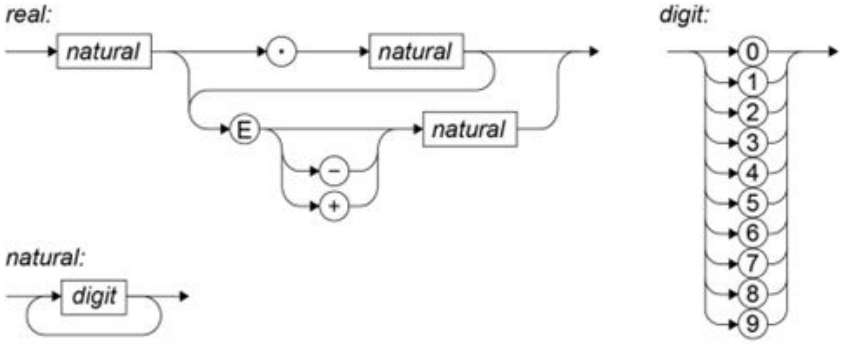
Statement type	String	Valid (Yes/No)
<number>	129.376	
<factor>	23 + 17	

(2 marks)

June 2017 Paper 1

0 1 In a particular programming language, the correct syntax for a real number, natural number and digit is defined by the syntax diagrams in **Figure 1**.

Figure 1



0 1 . 1 Write **Yes** or **No** in the unshaded cells in **Table 1** to identify whether or not the numbers listed in the table are valid real numbers which conform to the correct syntax for this language.

[3 marks]

Table 1

Real number	Valid? (Yes/No)
87.000	
97+12	
12.31E+12	

0 1 . 2 In Backus-Naur Form (BNF) the following production rule has been written to define a digit:

```
<digit> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
```

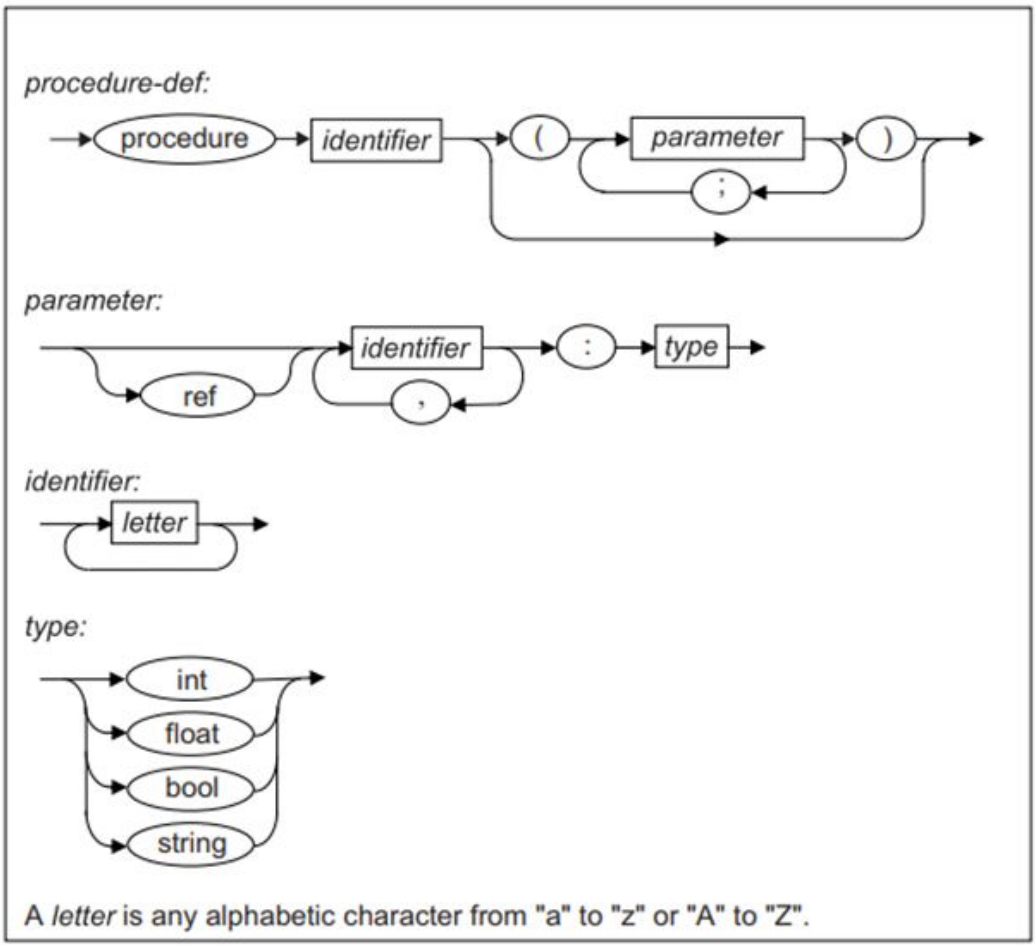
Write a BNF production rule to define a natural number that is equivalent to the definition in the syntax diagram in **Figure 1**.

[2 marks]

June 2013 Comp 3

1 In a particular programming language, the correct syntax for four different constructs is defined by the syntax diagrams in **Figure 1**.

Figure 1



A *letter* is any alphabetic character from "a" to "z" or "A" to "Z".

1 (a) For each row in the table below, write **Yes** or **No** in the empty column to identify whether or not the **Example** is a valid example of the listed **Construct**.

Construct	Example	Valid? (Yes/No)
identifier	`Player2name`	
parameter	`x,y:bool`	
procedure-def	`procedure square(s:real)`	
procedure-def	`procedure rect(w:int,h:int)`	

(4 marks)

1 (b) A student has written Backus-Naur Form (BNF) production rules that are supposed to define the same constructs as the syntax diagrams in **Figure 1**. Their BNF rules are shown in **Figure 2**.

Figure 2

```
<procedure-def>::= procedure <identifier> ( <paramlist> )
<paramlist>    ::= <parameter> | <parameter> ; <paramlist>
<parameter>    ::= <identlist> : <type> |
                   ref <identlist> : <type>
<identlist>    ::= <identifier> | <identifier> , <identlist>
<identifier>   ::= <letter> | <letter> <identifier>
<type>         ::= int | float | bool | char | string
```

A <letter> is any alphabetic character from "a" to "z" or "A" to "Z".

1 (b) (i) The BNF production rules in **Figure 2** contain two errors. These errors mean that they do not represent the same statement types as the syntax diagrams in **Figure 1**.

Describe the **two** errors.

Error 1: .....

.....

Error 2: .....

.....

(2 marks)

1 (b) (ii) The production rule for a <paramlist> is recursive.

Explain why recursion has been used in this production rule.

.....

.....

.....

.....

(1 mark)

Specimen Paper 1

Backus-Naur Form (BNF) can be used to define the rules of a language.

Figure 2 shows an attempt to write a set of BNF production rules to define a language of full names.

Figure 2

Note: underscores (_) have been used to denote spaces.
Note: rule numbers have been included but are not part of the BNF rules.

Rule number	
1	<fullname> ::= <title>_<name>_<endtitle> | <name> | <title>_<name> | <name>_<endtitle>
2	<title> ::= MRS | MS | MISS | MR | DR | SIR
3	<endtitle> ::= ESQUIRE | OBE | CBE
4	<name> ::= <word> | <name>_<word>
5	<word> ::= <char><word>
6	<char> ::= A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z

BNF can be used to define languages that are not possible to define using regular expressions. The language defined in **Figure 2** could not have been defined using regular expressions.

0 2 . 3 Complete **Table 2** below by writing either a '**Y**' for **Yes** or '**N**' for **No** in each row.

Table 2

Rule number (given in Figure 2)	Could be defined using a regular expression
1	
2	
3	
4	
5	
6	

Copy your answer in **Table 2** into the Electronic Answer Document.

[1 mark]

There is an error in rule 5 in **Figure 2** which means that no names are defined by the language.

0 2 . 4 Explain what is wrong with the production rule and rewrite the production rule so that the language does define some names – the names 'BEN D JONES', 'JO GOLOMBEK' and 'ALULIM' should all be defined.

[2 marks]