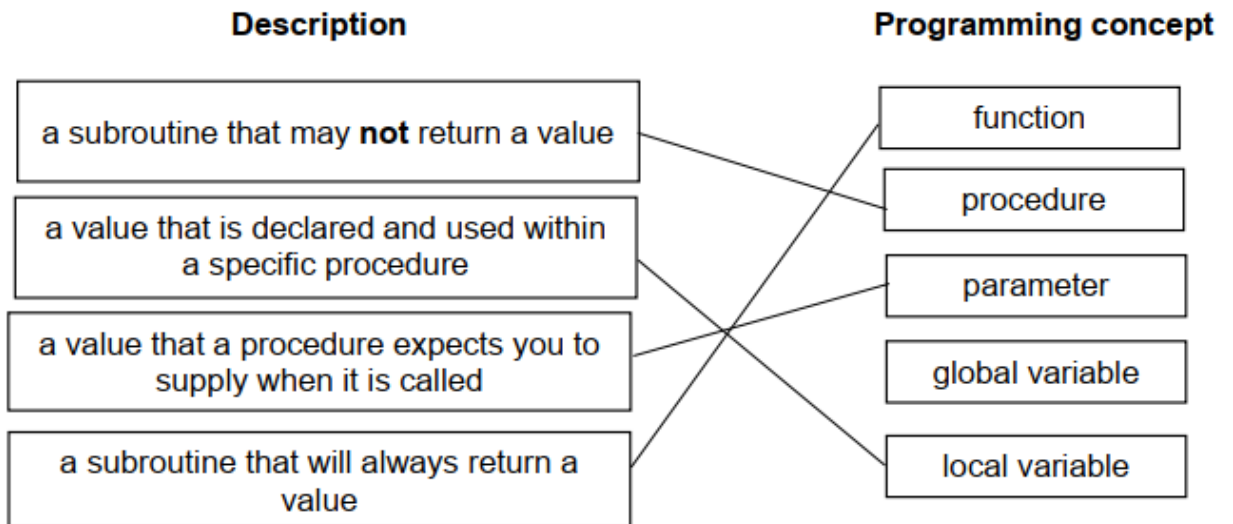


1. A)

One mark for each correct line



b)

One mark per mark point

- Correct key word - `CALL`
- Procedure name with correct parameters - `Average (25, 50)`

`CALL Average (25, 50)`

2.

one mark for first description **one** mark for matching difference **max four**

- local variables - scope is a defined block of code/subroutine/procedure/function
- global variables – scope is the whole program
- local variables - value cannot be changed elsewhere in the program
- global variables – value can be changed anywhere in the program

3.

```
PROCEDURE DisplayGreeting(Name:STRING)
    OUTPUT "Hello, ", Name, "!"
ENDPROCEDURE
```

```
OUTPUT "Enter your name:"
INPUT UserName
CALL DisplayGreeting(UserName)
```

4.

```
FUNCTION AddNumbers(Num1:INTEGER, Num2:INTEGER) RETURNS INTEGER
    RETURN Num1 + Num2
ENDFUNCTION
```

```
// Main program
OUTPUT "Enter the first number:"
INPUT Number1
OUTPUT "Enter the second number:"
INPUT Number2
```

```
Result ← AddNumbers(Number1, Number2)
OUTPUT "The sum is: ", Result
```

10

C

5.

Example definition:

```
FUNCTION IsDivisible(A : REAL, B : REAL) RETURNS BOOLEAN
    IF MOD(A, B) = 0
        THEN
            RETURN TRUE
        ELSE
            RETURN FALSE
    ENDIF
ENDFUNCTION
```

Example calling:

Result ← IsDivisible(M, N)

6.

Example definition:

```
FUNCTION MaxValue(A : INTEGER, B : INTEGER, C : INTEGER) RETURNS INTEGER
  IF A > B AND A > C
    THEN
      RETURN A
    ELSE
      IF B > A AND B > C
        THEN
          RETURN B
        ELSE
          RETURN C
      ENDIF
    ENDIF
  ENDFUNCTION
```

Example calling:

```
MaxNum ← MaxValue(X, Y, Z)
```

7.

```
PROCEDURE GreetUser(Name: STRING)
```

```
  OUTPUT "Hello, ", Name, "! Welcome to the program."
```

```
END PROCEDURE
```

```
CALL GreetUser("Alice")
```

8.

PROCEDURE CalculatePerimeter(Length: INTEGER, Width: INTEGER)

 Perimeter $\leftarrow$ 2 * (Length + Width)

 OUTPUT "The perimeter is ", Perimeter

END PROCEDURE

CALL CalculatePerimeter(8, 6)