Extension activity the nested IF .. THEN .. ELSE statement

Sometimes we have a situation where there are three or more outcomes to a decision.

Consider a student exam mark.
Less than 40 is a fail, 40-70 is a pass and over 70 is a distinction.

The teacher needs a program to read in the mark and output either ‘Fail’, ‘Pass’ or ‘Distinction’.

A nested IF statement is needed. This simply means that one IF statement occurs inside a second IF statement.

```
IF (mark < 40) THEN
  Write ‘FAIL’
ELSE
  IF (mark < 70) THEN
    Write ‘PASS’
  ELSE
    WRITE ‘DISTINCTION’
```

If the mark is less than 40 then write ‘FAIL’. So far this is like a regular IF statement. Now look at what happens if the mark is greater than or equal to 40. It is another IF statement nested inside the first IF statement. If the mark is less than 70, it must be between 40 and 70, then write ‘Pass’ otherwise write ‘Distinction’.

The Input-Process-Output (IPO) chart is:

Input	Processing	Output
mark	If mark < 40 grade is Fail else If mark < 70 grade is Pass Else grade is Distinction	grade

Caption: An IPO chart for the Grade algorithm.

The algorithm, Pseudocode, Pascal and Python programs.

Algorithm

```
Write ‘A simple program to grade a mark.’;
Write ‘Please give the mark.’
Get the mark
If the mark is less than 40 then
   Write ‘Fail’
Else
   If the mark is less than 70 then
      Write ‘Pass’
   Else
      Write ‘Distinction’
```

Pseudocode

```
WRITE ‘A simple program to grade a mark.’
WRITE ‘Please give the mark.’
READ the mark
IF the mark < 40 THEN
   WRITE ‘Fail’
ELSE
   IF the mark < 70 THEN
      WRITE ‘Pass’
   ELSE
      WRITE ‘Distinction’
```

Pascal program

```
PROGRAM grade;
VAR
mark : INTEGER;
BEGIN
   WRITELN ('A simple program to grade a mark');
   WRITELN ('Please give the mark.');
   READ (mark);
   WRITELN (mark);
   IF (mark < 40) THEN
      BEGIN
         WRITELN ('Fail');
      END
   ELSE
      BEGIN
            IF (mark < 70) THEN
               BEGIN
                 WRITELN ('Pass');
                END
            ELSE
                BEGIN
                  WRITELN ('Distinction');
                END
       END;
   WRITELN ('Press Enter to finish.');
   READLN;
END.
```

Python program

```
mark = 0
print ('A simple program to grade a mark.')
mark = input('Please give the mark ')
if int(mark) < 40:
  print('Fail')
else:
  if int(mark) < 40:
     print('Pass')
  else:
     print('Distinction')
```

User documentation:

This program determines a student’s grade depending on the mark scored. The input is the mark scored.
Limitations: The input must be a number.

Test data:
This is a simple program so test data will include a few examples of expected input values and a few examples of more extreme values. Test data values that do not meet this criteria the expected result will be a run-time error.

Input values	Reason	Expected Result	Actual result
55	typical values	Pass	Pass
75	typical values	Distinction	Distinction
0	extreme values	Fail	Fail
1000000	extreme values	Distinction	Distinction
two	invalid type	error	error 106

Caption: Test data for testing a simple grading program