22. Program Design using Structure Diagrams (Sequences)

As described in lesson 12, it is usual to break any program down into a number of smaller tasks (corresponding to paragraphs) before writing the detailed COBOL instructions.

When using the programming style describe in lesson 21 (using PERFORM), this break-down can most clearly be shown using *Structure Diagrams*.
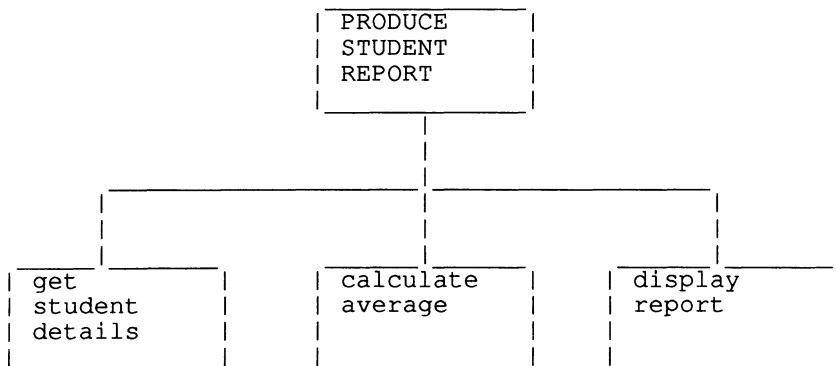
Example problem - Produce a Student Report.
Write a program to ask for a student's name and marks in Maths and English, calculate the average mark and display a report.

This program can be divided into a *sequence* of three main tasks:
1.   get the student's details;
2.   calculate the average mark;
3.   display the report.

The breakdown can be shown using a structure diagram as follows:

```
                              _____
                             | PRODUCE        |
                             | STUDENT        |
                             | REPORT         |
                             |_____|
                                     |
                                     |
                _____
               |                     |                      |
               |                     |                      |
          _____|_____          _____|_____           _____|_____
         | get        |        | calculate  |         | display    |
         | student    |        | average    |         | report     |
         | details    |        |            |         |            |
         |_____|        |_____|         |_____|
```

Note
1.   The overall name for the job is shown in the rectangle at the top of the diagram - the smaller tasks it splits into are shown under this in the order they are carried out (from left to right).

Exercises
1.   Produce Structure Diagrams for the exercises in lesson 12.
2.   What is meant by a sequence ?

25