EXAMPLE PROGRAM ATTRIBUTES REPORT

1. Requirements

To develop, using the Ada attributes facility, a test program to determine the range of the integer type and the range and precision of the floating point type available on a particular computer.

2. Design

Using a top down design approach (Table 1) we can identify, a top level task, two level 2 tasks and five level 3 tasks.

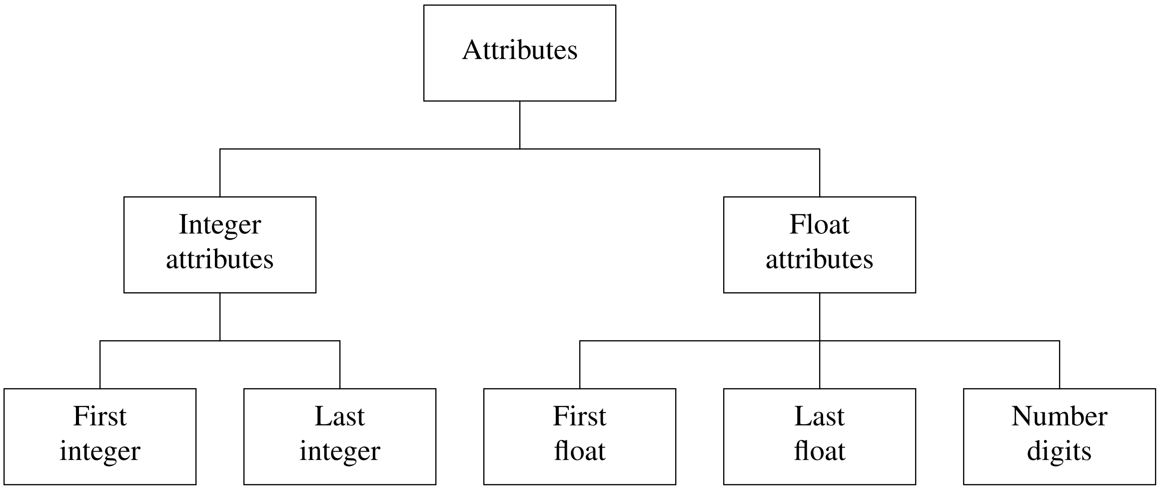
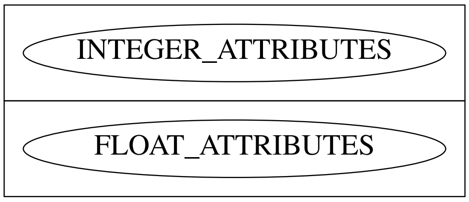


Table 1: *Top-down analysis*

The level 3 tasks identified in Table 1 are sufficiently trivial to warrant procedures of their own. Thus we will implement the design using two level 2 procedures (and a top level procedure). as follow:

1) `ATTRIBUTES` (top level procedure) - Calls to `INTEGER_ATTRIBUTES` and `FLOAT_ATTRIBUTES`.
2) `INTEGER_ATTRIBUTES` (level 2 procedure) - Output first and last integer.
3) `FLOAT_ATTRIBUTES` (level 2 procedure) - Output first and last float and number of digits.

Nassi-Shneiderman designs for these three procedures are given in Table 2.



ATTRIBUTES
(Level 1 Procedure)

Output first integer
Output last integer

INTEGER_ATTRIBUTES
(Level 2 Function)

Output first float
Output last float
Output number of digits

FLOAT_ATTRIBUTES
(Level 2 Function)

Table 2: *Nassi-Shneiderman Chart*

3. Implementation

Top down implementation commencing with level 1 top level procedure and the level 2. Final implementation will be as shown in Table 3:

```
-- ADA ATTRIBUTES
-- 6 August 1997
-- Frans Coenen
-- Dept Computer Science, University of Liverpool

with TEXT_IO;
use TEXT_IO;

procedure ATTRIBUTES is
        package INTEGER_INOUT is new INTEGER_IO(INTEGER);
        use INTEGER_INOUT;

------------------------------------------------------------
-- INTEGER TYPE ATTRIBUTES PROCEDURE
        procedure INTERGER_ATTRIBUTES is
        begin
                PUT("first  = ");
                PUT(INTEGER'FIRST);
                NEW_LINE;
                PUT("last   = ");
                PUT(INTEGER'LAST);
                NEW_LINE;
        end INTERGER_ATTRIBUTES;
```

Table 3: *Implementation*

```
-----------------------------------------------------------
-- FLOATING POINT TYPE ATTRIBUTES PROCEDURE
        procedure FLOAT_ATTRIBUTES is
                package FLOAT_INOUT is new FLOAT_IO(FLOAT);
                use FLOAT_INOUT;

        begin
                PUT("digits = ");
                PUT(FLOAT'DIGITS);
                NEW_LINE;
                PUT("small  = ");
                PUT(FLOAT'SMALL);
                NEW_LINE;
                PUT("large  = ");
                PUT(FLOAT'LARGE);
                NEW_LINE;
        end FLOAT_ATTRIBUTES;

-----------------------------------------------------------
-- TOP LEVEL PROCEDURE
begin
        INTERGER_ATTRIBUTES;
        FLOAT_ATTRIBUTES;
end ATTRIBUTES;
```

Table 3 Continued

4. Testing

The above code includes no input therefore a single execution will suffice to test the implementation. A screen dump illustrating the output produced is given in Table 4.

```
kuban-291 $ attributes
first  = -2147483648
last   =  2147483647
digits =          15
small  =  1.94469227433161E-62
large  =  2.57110087081438E+61
```

Table 4: *Test output*