EXAMPLE PROGRAM CYLINDER CALCULATION REPORT

1. Requirements

To produce a program that calculates the surface area and volume of a cylinder given its diameter and height. Assume "PI" is equivalent to 3.14 and that the diameter and height are input by the user as a floating point number. Area of a cylinder = (2 x AreaOfEnd) + (AreaOfSide) = (2 x (PI x (diameter/2)^2)) + (height x PI x diameter), Volume of a cylinder = AreaOfEnd x height = (PI x (diameter/2)^2) x height.

2. Design

Using a top-down hierarchical decomposition approach we can identify the break-down of operations shown in Table 1.

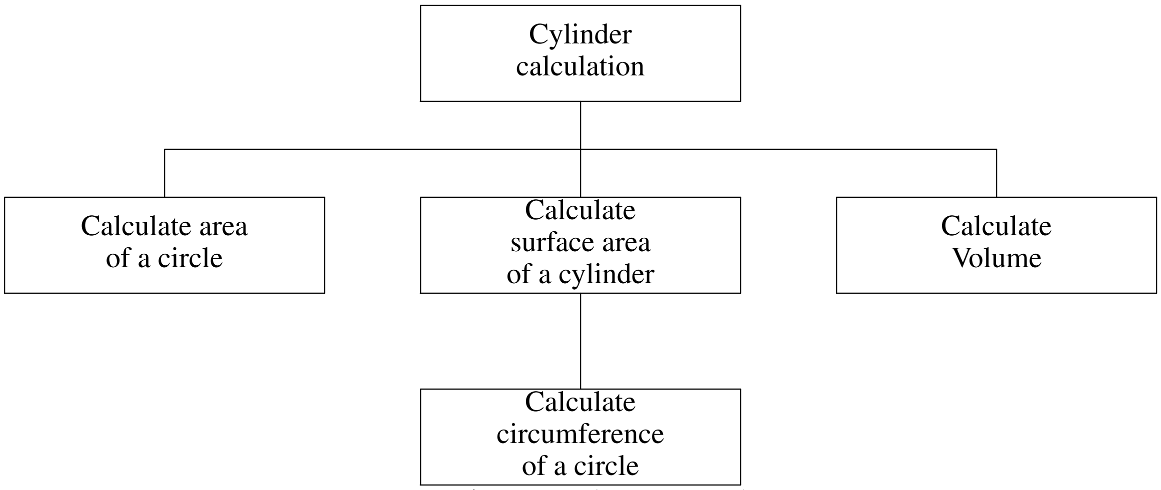


Table 1: *Top-down analysis*

From the above we can identify the following procedures and functions:

1) **CYLINDER_CALC** (Top level procedure): Input data and calls to **AREA_OF_CIRCLE**, **AREA_OF_CYLINDER** and **VOLUME_OF_CYLINDER**.

Name	Description	Type	Range
PI	Global constant	FLOAT	3.15
DIAMETER	Global variable	FLOAT	Default
HEIGHT	Global variable	FLOAT	Default
AREA	Global variable	FLOAT	Default

2) **AREA_OF_CIRCLE** (level 2 function) - calculate and return area of circle.

Name	Description	Type	Range
RADIUS	Formal parameter	FLOAT	Default

3) **AREA_OF_CYLINDER** (level 2 procedure) - calculate and output area of cylinder. Call to **CIRCUM_OF_CIRCLE**.

Name	Description	Type	Range
DIAMETER	Formal parameter	FLOAT	Default
HEIGHT	Formal parameter	FLOAT	Default
AREA	Formal parameter	FLOAT	Default
CIRCUM	Local variable	FLOAT	Default

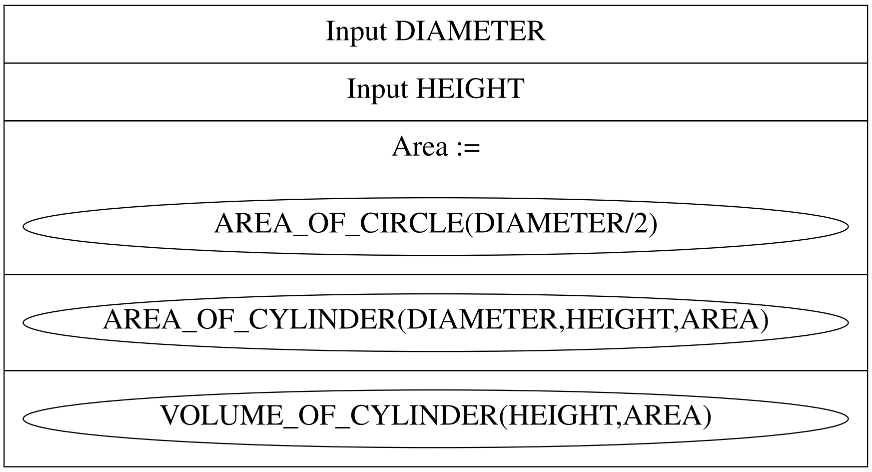
4) **VOLUME_OF_CYLINDER** (level 2 procedure) - calculate and output volume of cylinder.

Name	Description	Type	Range
NAME	USAGE	TYPE	RANGE
HEIGHT	Formal parameter	FLOAT	Default
AREA	Formal parameter	FLOAT	Default

5) **CIRCUM_OF_CIRCLE** (level 3 function) - calculate and return circumference of circle.

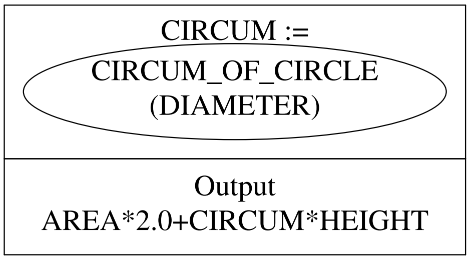
Name	Description	Type	Range
NAME	USAGE	TYPE	RANGE
DIAMETER	Formal parameter	FLOAT	Default

The design detail required for each of these is given by the Nassi-Shneiderman charts presented in Table 2. Note that design involves no input, only output.



CYLINDER_CALC
(Top level procedure)

AREA_OF_CIRCLE
(Level 2 Function)



AREA_OF_CYLINDER_AREA
(Level 2 Procedure)

Output AREA*HEIGHT	Return PI*DIAMETER
CYLINDER_VOLUME (Level 2 Procedure)	CIRCUM_OF_CIRCLE (Level 3 Function)

Table 2: *Nassi-Shneiderman Charts*

3. Implementation

Implementation is as shown in Table 3.

```
-- CYLINDER CALCULATION
-- 4 August 1997
-- Frans Coenen
-- Dept Computer Science, University of Liverpool

with TEXT_IO;
use TEXT_IO;

procedure CYLINDER_CALC is
        package FLOAT_INOUT is new FLOAT_IO(FLOAT);
        use FLOAT_INOUT;
        PI: CONSTANT FLOAT := 3.14;
        DIAMETER, HEIGHT, AREA: FLOAT;

-----------------------------------------------------
-- AREA OF CIRCLE FUNCTION
        function AREA_OF_CIRCLE(RADIUS: FLOAT)
                        return FLOAT is
        begin
                RETURN(PI*RADIUS**2);
        end AREA_OF_CIRCLE;
```

Table 3: *Implementation*

```
-----------------------------------------------------
-- AREA OF CYLINDER PROCEDURE
        procedure AREA_OF_CYLINDER(DIAMETER, HEIGHT, AREA: FLOAT) is
        CIRCUM: FLOAT;

        ---------------------------------------------
        -- CIRCUMFERENCE OF CIRCLE FUNCTION
                function CIRCUM_OF_CIRCLE (DIAMETER: FLOAT)
                                return FLOAT is
                begin
                        RETURN(PI*DIAMETER);
                end CIRCUM_OF_CIRCLE;
        ---------------------------------------------
        begin
                CIRCUM:= CIRCUM_OF_CIRCLE(DIAMETER);
                PUT("Area of cylinder is: ");
                PUT(AREA*2.0+CIRCUM*HEIGHT,FORE=>3,
                                AFT=>4,EXP =>0);
                NEW_LINE;
        end  AREA_OF_CYLINDER;

-----------------------------------------------------
-- VOLUME OF CYLINDER FUNCTION
        procedure VOLUME_OF_CYLINDER(HEIGHT, AREA: FLOAT) is
        begin
                PUT("Volume of cylinder is: ");
                PUT(AREA*HEIGHT,FORE=>3,AFT=>4,
                                EXP =>0);
                NEW_LINE;
        end VOLUME_OF_CYLINDER;

-----------------------------------------------------
-- TOP LEVEL PROCEDURE
begin
        PUT_LINE("Input diameter: ");
        GET(DIAMETER);
        PUT_LINE("Input height: ");
        GET(HEIGHT);
        AREA := AREA_OF_CIRCLE(DIAMETER/2.0);
        AREA_OF_CYLINDER(DIAMETER, HEIGHT, AREA);
        VOLUME_OF_CYLINDER(HEIGHT, AREA);
end CYLINDER_CALC;
```

Table 3 continued

4. Testing

Arithmetic testing Test using all possible combinations of negative, positive and zero input values for the `DIAMETER` and `HEIGHT` user inputs as shown in the Table 4. Note that the code allows input of negative numbers. This is likely to be undesirable and this a return to the requirements specification should be instigated and adjustments made in the light of the procurers wishes with respect to negative results.

TEST CASE		EXPECTED RESULT	
DIAMETER	HEIGHT	AREA	VOLUME
0.0	0.0	0.0000	0.0000
0.0	10.0	0.0000	0.0000
0.0	-10.0	0.0000	0.0000
5.0	0.0	39.2500	0.0000
5.0	10.0	196.2500	196.2500
5.0	-10.0	-117.7500	-196.2500
-5.0	0.0	-39.2500	0.0000
-5.0	10.0	-117.7500	196.2500
-5.0	-10.0	196.2500	-196.2500

Table 4: *Test cases*

The result of running the above test cases is given in Table 5.

```
kuban-367 $ cylinder_calc
Input diameter:
0.0
Input height:
0.0
Area of cylinder is:   0.0000
Volume of cylinder is:   0.0000

kuban-368 $ cylinder_calc
Input diameter:
0.0
Input height:
10.0
Area of cylinder is:   0.0000
Volume of cylinder is:   0.0000

kuban-369 $ cylinder_calc
Input diameter:
0.0
Input height:
-10.0
Area of cylinder is:   0.0000
Volume of cylinder is:   0.0000

kuban-370 $ cylinder_calc
Input diameter:
5.0
Input height:
0.0
Area of cylinder is:  39.2500
Volume of cylinder is:   0.0000
```

Table 5: *Test output forth arithmetic test cases*

```
kuban-371 $ cylinder_calc
Input diameter:
5.0
Input height:
10.0
Area of cylinder is: 196.2500
Volume of cylinder is: 196.2500

kuban-372 $ cylinder_calc
Input diameter:
5.0
Input height:
-10.0
Area of cylinder is: -117.7500
Volume of cylinder is: -196.2500

kuban-373 $ cylinder_calc
Input diameter:
-5.0
Input height:
0.0
Area of cylinder is: -39.2500
Volume of cylinder is:   0.0000

kuban-377 $ cylinder_calc
Input diameter:
-5.0
Input height:
10.0
Area of cylinder is: -117.7500
Volume of cylinder is: 196.2500

kuban-378 $ cylinder_calc
Input diameter:
-5.0
Input height:
-10.0
Area of cylinder is: 196.2500
Volume of cylinder is: -196.2500
```

Table 5 Continued

Data validation testing: Provide input data with incorrect syntax, both blatant and subtle. Suggested test cases are given in Table 6 (output in table 7). The '*' symbol indicates that we do not expect to input a second value as we expect the code to fail.</P>

TEST CASE		EXPECTED RESULT	
DIAMETER	HEIGHT	AREA	VOLUME
5	*	FLOAT_INPUT_ERROR_PERIOD_NOT_READ	
5.e	*	FLOAT_INPUT_ERROR_PERIOD_NOT_READ	
5.0	10	FLOAT_INPUT_ERROR_PERIOD_NOT_READ	
5.0	1e.0	FLOAT_INPUT_ERROR_PERIOD_NOT_READ	

Table 6: *Data validation test cases*

```
kuban-165 $ cylinder_calc
Input diameter:
5
Ada-runtime: Exception FLOAT_INPUT_ERROR_PERIOD_NOT_READ propagated
out of main.

kuban-167 $ cylinder_calc
Input diameter:
5.e
Ada-runtime: Exception FLOAT_INPUT_ERROR_DIGIT_NOT_READ propagated
out of main.

kuban-168 $ cylinder_calc
Input diameter:
5.0
Input height:
10
Ada-runtime: Exception FLOAT_INPUT_ERROR_PERIOD_NOT_READ propagated
out of main.

kuban-169 $ cylinder_calc
Input diameter:
5.0
Input height:
1e.0
Ada-runtime: Exception FLOAT_INPUT_ERROR_PERIOD_NOT_READ propagated
out of main.
```

Table 7: *Test output from data validation test cases*