

FORM 4 PASCAL PROGRAMMING**Unit 3: Arithmetic Operations & Assignment Statements**

March, 01

3.1. ARITHMETIC OPERATIONS (算術運作)

- Several integer operations are summarized as follows:

<i>Operation</i>	<i>Purpose</i>	<i>Type of Resultant</i>
+a	a	INTEGER
-a	negation of a	INTEGER
a + b	sum of a and b	INTEGER
a - b	difference of a and b	INTEGER
a * b	product of a and b	INTEGER
a / b	quotient of division of a by b	☆ REAL
a div b	<i>integral quotient</i> of division of a by b	INTEGER
a mod b	remainder of division of a by b	INTEGER

- Example: Suppose that a and b are integer-type variables where a=13 and b=5

<i>Expression</i>	<i>Value</i>
a + b	
a - b	
a * b	
a / b	
a div b	
a mod b	

- a number of type REAL can be expressed either *decimal* notation or in *scientific* notation.

e.g. 125.2 (decimal notation)
 1.252E+02 (scientific notation)

- REAL operations are summarized as follows:

<i>Operation</i>	<i>Purpose</i>	<i>Type of Resultant</i>
+R	R	REAL
-R	negation of R	
R + S	sum of r and s	
R - S	difference of r and s	
R * S	product of r and s	
R / S	quotient of division of r by s	
TRUNC (R)	integral part of R	
ROUND (R)	the closest integer of R	

- Example: Suppose that *c* and *d* are real-type variables whose assigned values are 12.4 and 0.5 respectively. Several real expressions and their corresponding values are shown below:

<i>Expression</i>	<i>Value</i>
$c + d$	
$c - d$	
$c * d$	
c / d	
round(<i>c</i>)	
trunc(<i>c</i>)	
round(<i>d</i>)	
truc(<i>d</i>)	

- **Notes:**

The two division operators (/ and DIV) and the MOD operator require that the *second operand be non-zero*;

If one operand is of the type integer, and the other is of the type REAL, then the resulting value will always be of the type **REAL**.

3.2. STANDARD ARITHMETIC FUNCTIONS

- Pascal contains a number of standard functions that are used with various basic data types:

Function	Purpose	Type Of Parameter 參數 (X)	Type Of Resultant 結果
ABS (X)	Absolute value of X.	integer or real	same as X
SIN (X)	Sine of X (X in radians).	integer or real	real
COS (X)	Cosine of X (X in radians).	integer or real	real
SQR (X)	Square of X.	integer or real	same as X
SQRT (X)	Square root of X.	integer or real	real
PRED (X)	Predecessor of X.	integer, char or Boolean	same as X
SUCC (X)	Successor of X.	integer, char or Boolean	same as X
TRUNC (X)	Truncate X (i.e. drop the decimal part of X).	real	integer
ROUND (X)	Round X (determine the closest integer)	real	integer
INT (X)	Integral part of X.	real	real
FRAC (X)	Fractional part of X.	real	real
PI	Value of π (3.14159265...)	(none)	real

- Examples:

PRED (3) =

SUCC (3) =

ROUND (2 . 3) =

TRUNC (2 . 3) =

ROUND (3 . 7) =

TRUNC (3 . 7) =

ROUND (-1 . 8) =

TRUNC (-1 . 8) =

ROUND (-6 . 1) =

TRUNC (-6 . 1) =

INT (3 . 7) =

FRAC (3 . 7) =

3.3. MATHEMATICAL EXPRESSION → PASCAL EXPRESSION

<i>Mathematics Expression</i>	<i>Pascal Expression</i>
$Ax^2 + Bx + C$	
$(x+1)(x+7)$	
$3abc - 6$	
$\frac{A + 2B + 1}{AB}$	
$(xy + z)^2$	

3.4. ASSIGNMENT STATEMENTS

- **Purpose:** to assign a data item into a variable
- **Syntax:**

```
variable := data_item;
```

where `data_item` can be constant, variable, function reference or expression

- **Examples:**

```
(i)   total           := 100;
(ii)  result          := 2*a + 3*b;
(iii) name            := 'Cheung Tai-chin';
(iv)  number2         := number1;
(v)   remainder       := 60 mod 7;
(vi)  area             := base * height/2;
(vii) square_area    := sqr(length);
(viii) count          := count + 1;
```

- **Notes:**

Variable on the left-hand side of the assignment;

':= ' means "gets the value of";

The data item must be of the *same type* as the variable identifier (EXCEPTION: an INTEGER data item can be assigned to a REAL variable);

end of unit 3