CE 150 COMPUTER PROGRAMMING

EXPRESSIONS & OPERATORS

1

EXPRESSIONS

- An expression can be a string or numeric constant, a variable, or a single (only one) value obtained by combining constants, variables, and other expressions with operators.

2

OPERATORS

- Operators perform mathematical or logical operations on values. The operators provided by BASIC can be divided into five categories:

3

OPERATORS (Continued)

1. Arithmetic Operators
2. Relational Operators
3. Logical Operators
4. Functional Operators
5. String Operators

Numeric

STRING

4

ARITHMETIC OPERATORS

- These operators perform arithmetic operation.
- There are seven operators provided by BASIC.

5

ARITHMETIC OPERATORS

OPERATOR	OPERATION	EXAMPLE
^	Exponentiation	X^Y X^Y
-	Negation	-X
* /	Multiplication Division	X*Y XxY X/Y X%Y
\	Integer Division	X\Y
MOD	Modulo Arithmetic	X MOD Y
+ , -	Addition, Subtraction	X+Y or X-Y

6

ARITHMETIC OPERATORS

- The order in which the operators are listed in Table is order of precedence.
- Although most of these operations probably look familiar to you, two of them may seem a bit unfamiliar:
- Integer Division & Modulo Arithmetic

7

\ Integer Division

- Integer division is denoted by backslash. The operands are rounded to long Integers (if within the range) before the division is performed.
- The quotient is then truncated to an integer.

8

\ Integer Division

Example

- A=10\4 gives 2.5 which then truncates to 2. Thus A=2
- B=25.52\6.99 The operation is performed as follows:
- 26\7 =3.7 ➔ 3
- Thus B=3

9

MOD

Example

- Modulo arithmetic is denoted by MOD. It gives the integer value that is the remainder of an integer division:
- A=7 MOD 4 will give A=3
- B=25.68 MOD 6.88 gives B=5
- (remember BASIC rounds when converting to Integer)

10

RELATIONAL Operators

- Relational operators compare two values. The values may be either both numeric, or both string.
- The result of comparison is either TRUE (-1) or FALSE(0).
- This results is then used in making decisions regarding flow of program.

11

RELATIONAL Operators

OPERATOR	RELATION TESTED	EXAMPLE
=	Equality (Equal TO)	A=B
<> or ><	Inequality (Not =)	A<>B
<	Less Than	X < Y
>	Greater Than	A > B
<= or =<	Not Greater Than	A <= X
>= or =>	Not Less Than	Z => B

12

ARITH and RELAT Operators

- When arithmetic operators are combined with relational operators in one expression, the arithmetic is performed first:
- X+Y < (T+1)/Z
- First X+Y and (T+1)/Z are obtained and then comparison is made.

Example

- 5 < 2 will yield False i.e. 0
- 5 < 10 will yield True I.e. –1

STRING COMPARISON

- The two strings are compared by taking one character from each and comparing the ASCII codes.
- IF all the ASCII codes are same the two strings are equal.
- Otherwise as soon as the ASCII codes differ, the string with the lower code is less than the other.
- If during comparison the end of one string is reached, it is less than other.

STRING COMPARISON EXAMPLE

- "AA" < "AB"
- "ABCX"="ABCX"
- "ABCX" < "ABCX "
- "ABCx" > "ABCXYZabc"
- "kg" > "KG"
- "718" > "12345678"
- All the above expressions give TRUE.

LOGICAL Operators

- Logical operators perform logical or Boolean operations on numeric values.
- Just as the relational operators are used to make decisions regarding program flow, logical operators are usually used to connect two or more relations and return a true or false value to be used in a decision.

LOGICAL Operators

The six logical operators are

- NOT (Complement)
- AND (Conjunction)
- OR (Disjunction) (Inclusive OR)
- XOR (Exclusive OR)
- EQV (Equivalence)
- IMP (Implication)

LOGICAL Operators

X	Y	NOT X	X AND Y	X OR Y	X XOR Y	X EQV Y	X IMP Y
T	**T**	F	T	T	F	T	T
T	F	F	F	T	T	F	F
F	**T**	T	F	T	T	F	T
F	F	T	F	F	F	T	T

19

LOGICAL Operators EXAMPLES

- IF HE > 60 and SHE < 20 will return a true if value of HE is greater than 60 and that of SHE is less than 20.
- IF A > 4 OR B < 0 will yield true if either A is greater than 4, or B is less than zero, or both.
- NOT (P=-1) will return true if P <> -1

20

LOGICAL Operators

- The operands of logical operators are converted to integers or long (in the range). If they are out of range of long an error occurs. Otherwise a bit wise comparison is made and 1 is considered true and 0 as false.

21

LOGICAL Operators EXAMPLES

- A=63 AND 16
- 63= 0000 0000 0111 1111
- 16= 0000 0000 0000 0100
- 16= 0000 0000 0000 0100
- Thus A is set to 16.

22

LOGICAL Operators EXAMPLES

- A=63 OR 16
- 63= 0000 0000 0111 1111
- 16= 0000 0000 0000 0100
- 63= 0000 0000 0111 1111
- Thus A is set to 63.

23

LOGICAL Operators EXAMPLES

- C= 4 OR 2
- 4= 0000 0000 0000 0100
- 2= 0000 0000 0000 0010
- 6= 0000 0000 0000 0110
- Thus C=6

24

LOGICAL Operators
EXAMPLES

- X=2
- (NOT X)+1 will give you –2
- X=2 = 0000 0000 0000 0010
- NOT X = 1111 1111 1111 1101
- 1+
- Not X+1 1111 1111 1111 1110 =-2

25

LOGICAL Operators
EXAMPLES

- X=-5
- (NOT X)+1 will give you 5
- X=-5 = 1111 111 1011
- N 0000 0000 0000 0100
- 1+
- Not X+1 0000 0000 0000 0101 =5

NOT (A) +1 converts A to -A

26

LOGICAL Operators
EXAMPLES

- 4 XOR 5 gives 1
- 0000 0000 0000 0100

XOR

- 0000 0000 0000 0101
- 0000 0000 0000 0001

27

LOGICAL Operators
EXAMPLES

- 7 XOR 12 gives 11
- 0000 0000 0000 0111

XOR

- 0000 0000 0000 1100
- 0000 0000 0000 1011 =11

28

LOGICAL Operators
EXAMPLES

- 207 XOR 120 gives 183
- 0000 0000 1100 1111

XOR

- 0000 0000 0111 1000
- 0000 0000 1011 0111 =183

29

LOGICAL Operators
EXAMPLES

- 207 EQV 120 gives -184
- 0000 0000 1100 1111

EQV

- 0000 0000 0111 1000
- 1111 1111 0100 1000 = -184

30

LOGICAL Operators

EXAMPLES

- 207 IMP 120 gives -136
- 0000 0000 1100 1111

IMP

- 0000 0000 0111 1000
- 1111 1111 0111 1000 = -136

31

FUNCTIONAL Operators

- A function is used like a variable in an expression o call a predetermined operation that is to be performed on one or more operands.
- SIN, LOG, SQR are examples
- BASIC language provides a large number of functions like the above three.

32

Operators

ORDER OF EXECUTION

- The numeric operators have been discussed in 4 categories and in each category the precedence of operations within each category was indicated in the discussion of the category.
- The summary is given next:

33

Operators

ORDER OF EXECUTION

- FUNCTION are evaluated first.
- Arithmetic operations are evaluated next in the order (from left to right)

1. ^
2. - (Negation)
3. * /
4. \
5. MOD
6. + -

34

Operators

ORDER OF EXECUTION

- Relational operations are done next. L-R
- Logical operations are done last in the order:

1. NOT
2. AND
3. OR
4. XOR
5. EQV
6. IMP

35

Operators

ORDER OF EXECUTION

- Operations at the same level in the list are done in left to right order.
- To change the order of precedence use the parenthesis.
- Operations within parentheses are performed first.
- Inside the parentheses the usual order of operations is maintained.

36

EXAMPLES

ORDER OF EXECUTION

- X+2Y → X+Y*2
- $X+\frac{Y}{Z}$ → X+Y/Z
- $\frac{XY}{Z}$ → X*Y/Z
- $\frac{X+Y}{Z}$ → (X+Y)/Z
- $(X^2)^Y$ → X^2^Y
- X^{Y^Z} → X^(Y^Z)

37

EXAMPLES

ORDER OF EXECUTION

- X(-Y) → X^(-Y) or X^-Y
- (-b+sqr(b^2-4*a*c))/2/a gives

$$\frac{-b+\sqrt{b^2-4ac}}{2a}$$

38

EXAMPLES

ORDER OF EXECUTION

- Two consecutive operators must be separated by parentheses.
- Exceptions to this rule are * -, * +, ^ -, and ^ +.
- X*-Y is valid as X*(-Y) and
- X^-Z is also valid as X^(-Z)

39