CS 241
Control Structures

Christopher A. Gantz

SPS Undergraduate Program
Regis University
cgantz@regis.edu

Lecture 9

Overview of CS 241: Statements, Arithmetic Elements & Operations

Christopher A. Gantz
School of Professional Studies
Regis University
cgantz@regis.edu

Overview

- Reading
  - Nance textbook Pages 59-76 and 91-102
- Arithmetic Elements & Operations
  - Statements and Expressions
  - Integer Operations & Overflow
  - Mixed Mode Expressions
  - Standard Functions
  - Variables & Constants
  - Assignment Statements

Pascal Statements

- A Pascal statement is the basic unit of exexecution.
- There are two types of Pascal statements
  - Declaration statements
  - Processing/Executable statements
- Processing statements can be
  - A simple statement (i.e. A single action)
  - A compound statement (i.e. Collection of statements between BEGIN and END delimiters

Pascal Statements (Cont)

- Where an executable statement is comprised of an expression
- An expression can be:
  - A constant value
  - A variable
  - A sub-expression
- The expression has to be of a specific data type, such as integer, real, or character

Arithmetic Expressions

- An expression is an arbitrary combination of sub-expressions containing values and variables in order to produce a single typed value
- Where sub-expressions are combined with arithmetic symbols such as:
  - + addition operator
  - - subtraction operator
  - * multiplication operator
- Integer arithmetic operations are operations that are applied to data of type integer

Arithmetic Expressions (Cont)

- Expression examples
  - 5 * 7
  - SumOfdata / numOfData
  - sqrt(b*b – 4*a*c)
  - Avg := (1 + 2 + 3 + 4 +5) / 5
  - 18 DIV 2
- Expressions must be evaluated with respect to ***precedence rules*** which define the specific order of priority in which numeric operations are executed

Integer Operations (Cont)

- Special integer operations
  - MOD - integer modulus operator
  - DIV - integer division operator
- Order/precedence of integer operations
  - ( )
  - *, MOD, DIV
  - +, -
- Integer operations are expected to generate integer results, therefore the MOD or DIV operators must be utilized for integer division

Integer Overflow

- Integer operations have to be careful of integer overflow which occurs when the operation produces a value outside of the range of (-maxint-1, maxint)

Real Operations

- Basic real operations
  - + Addition
  - - Subtraction
  - * Multiplication
  - / Division
- Order/precedence of real operations
  - ( )
  - *, /
  - +, -

Mixed Mode Expressions

- Expression that contains both integer and real sub-expressions, i.e. Mixed mode
- Always results in the value of the expression being of data type real
- Example
  - Ratio := 7.0 / 5;
- Note: if two integers are divide with the '/' operator the resulting data type is real

Mixed Mode Expressions (Cont)

- Note: DO NOT USE MOD or DIV with real data typed sub-expressions

Standard Functions

- Pascal provides standard (built-in) function for certain standard functions
- Pascal functions are invoked(i.e. called) within a program by the following abstract(BNF) syntax:
    - <function name>(<arguments>);
- Where arguments are either constant values or a variable containing an assigned value of the correct data type

Numeric Standard Functions

- sqr(<real expr | integer expr>) --> real | integer - the square function
- sqrt(<real expr | integer expr>) --> real | integer - the square root function
- abs(<real expr | integer expr>) --> real | integer - the absolute value function
-

Numeric Standard Functions

- round(<real expr>) --> integer - the round function
- trunc(<real expr>) --> integer - the truncate function
- Examples
  - Sqr(2) == 4
  - Abs(-5) == 5
-

Character Standard Functions

- An ordinal data type is a data type that is ordered with respect to the set of integers
- The standard basic ordinal type is the char data type, specifcally the ASCII character set
    - See page 95 of text
  - ord(<argument>) --> integer value - the ordinal function
  - pred(<argument>)--> ordinal value - the predecessor function

Character Standard Functions (Cont)

-
  - succ(<argument>) --> ordinal value - the successor function
  - chr(<argument>) --> character - the character function
-

Pascal Variables & Constants

- A memory location is a storage element that has an address in the computer memory architecture
- A variable is memory location that is mapped to a valid identifier that can be changed
  - E.g. VAR index: integer;
- A constant is a memory location that is mapped to a valid identifier that can NOT be changed
  - E.g. CONST PI = 3.145;

Assignment Statements

- <variable name> := <expression> ;
- Both the variable and the expression must be of the same type, unless it is a mixed mode expression
- := is the assignment operator and is different from the equality/equals operator =
-