**Name of variables:**

Cannot use C++ keywords/reserved words

– Is case-sensitive

– Must start with either a letter or \_,

– The remaining characters must all be letters, digits, or underscore

**Assignment Rules**

Cannot place values of one type into variable of another type

Example

int myVal = 3.14;

– Only the integer part is taken

* Called implicit or automatic type conversion

Escape Sequences – Cont.

**Post-increment**

– Uses current value of a variable, Then increment it

**Pre-increment**

• Increment value first, Then use the new value

• Usage is defined as whatever context variable is currently in

• No difference if along in statement

Old Approach – Attach a type keyword to a variable • E.g. double (i); or (double) i: – May not be supported in the future

New Approach

– Syntax:

• **static\_cast <type> (expression);**

– Example:

• doubleVar = static\_cast <double> (intVal1)/intVal2;

Solution

– The Library functions convert other types to and from characters

• Example:

– The stream contains 45, two characters

– cin >> i; converts the two characters into an integer

* And stores it in a integer variable i.

Magic formula for decimal numbers:

cout.setf(ios::fixed);

cout.setf(ios::showpoint);

cout.precision(2);

cin >> variable;

• Semantics:

– Waits on-screen for keyboard entry

– Value entered at keyboard is assigned to variable

• >> is aware of various data types

• Multiple inputs can be cascaded to one input

cin >> number1 >> number 2;

String can be used by placing the following two lines

in the beginning of the code:

#include<string>

using namespace std;

• A string can hold zero or more characters

• String can be treated in the same way as other built-in

data types

– Declaration

string myName = “Yutao”; or

string myname(“Yutao”);

– Assignment

myName = “Yutao He”;

cin >> name; // suppose you type in “Yutao He” • Because it ignores initial whitespaces and will stop reading when it sees more whitespace • Solution - use getline() function string name; getline(cin, name);

Syntax

string1.length()

• Semantics

– Return the number of characters in string string1

– The returned value is an integer type int

• Note:

* You cannot set the string length

Namespace defines: – A set of collections of name definitions • For now we only use namespace std – It contains all standard library functions we need

Only “get” one statement per branch

• Must use compound statement {} for multiples

– aka block statement

• Each block should have block statements

– Even if there is only one statement

– Enhance the readability and specify the scope