What Data Do We Have?

Today

- On Wednesday I showed you a C++ program that converts distances from miles to kilometers
- What are the main components of that program?
- Today we will
  - learn how C++ stores data
  - Some of the different types of data that C++ can store

Declaration Statements

- The declaration statements in the program are
  - const double KM_PER_MILE = 1.609;
  - double miles;
  - double kms;
- With the above statements I am declaring three things
  - KM_PER_MILE to store the conversion rate that never changes
  - miles to store the distance in miles as given by the user
  - kms to store the distance in kilometers as calculated by the program

Variable Declaration

- Variables are declared by stating
  - Type of data (data type)
  - Name to identify the variable (identifier)
  - Semicolon (;)
    - data-type identifier;
    - double miles;

Variable Declaration

- If there is more than one variable of a single data type then you
  - State the data type
  - List the variable identifiers (names) separated by commas
    - data-type identifier1, identifier2;
    - double miles, kms;
Data types and Identifiers

• Data types
  o C++ can store many different types of data
  o A data type also defines what operations can be performed on data of that type
  o We will start with the three primitive data types
    - int (integer numbers)
    - double (real numbers)
    - char (characters)
  o These data types must be in lower case

• Identifiers
  o Valid variable names in C++

int

• The int data type is used to store integer numbers, both positive and negative
• int’s are finite (why?), i.e. they have a limited range that is implementation dependent
• Examples of int’s are: 123, -23, 0, 2352
• An int without a sign (+ or -) is assumed to be positive
• 2,353 is not an int, 2353 is an int
• What operations can be performed on integers?

double

• The double data type is used to store real numbers, both positive and negative
• Real numbers can contain fractional parts
• double’s are finite
• Examples of double’s are: 1.0, -2.3, -3, 12E5, -1E-2
• A double without a sign (+ or -) is assumed to be positive
• 2,353.99 is not a double, 2353.99 is a double

char

• The char data type is used to store single characters (letters, digits, special characters)
• char values are enclosed in single quotes
• Examples of char’s are: ‘A’, ‘a’, ‘*’, ‘2’, ‘$’

Examples

• Remember, the format for declaring variables is:
  o data-type identifier;
• You can declare variables of the different data types as follows
  o int num1;
  o double num2;
  o char letter;

Identifiers

• C++ does place limits on what names you can call your variables
• Rules
  1. Identifiers must begin with a letter or an underscore
  2. Identifiers must consist of letters, numbers and underscore, nothing else
  3. Identifiers cannot be a reserved keyword
Reserved Keywords

- These are words that are reserved by C++ to implement various features
- Examples of keywords that we have seen so far are int, double, const, return
- A list of C++ keywords can be found on page 75 of your textbook

Identifiers

- Identifiers are case sensitive
  - int num1;
  - int Num1;
  - num1 and Num1 are different variables
- You should always try to use meaningful variable names
  - If you have a variable that represents the width, then call it width not w

Identifiers

- Which of the following declarations are invalid and why?
  - char Letter1;
  - char lletter;
  - double inches, kms;
  - double inches*num;
  - int joe's;
  - Int cent_per_inch;
  - double two-dimensional;
  - char hello;
  - int return;
  - size int;

Variable Declarations

- All the variable declarations that we have seen are of the form
  - data-type identifier;
- This form declares a variable of the specific type, gives it the specific name (identifier) and allocates the space in memory to store the value of this variable
- However, no value has been assigned to this variable as yet

Variable Declarations

- When declaring multiple variables of the same type, it is preferable to place each variable declaration on a line along with a comment specifying the use of the variable
  - double miles; // Distance in miles from user
  - double kms; // Distance in kms

Constant Declarations

- Constants are declared by stating
  - const
  - Type of data (data type)
  - Name to identify the variable (identifier)
  - =
  - Value assigned to this constant that will never change
  - Semicolon (;)
  - const double KM_PER_MILE = 1.609;
Constant Declarations

• Constant values *never* change
  o KM_PER_MILE will always be 1.609

• In C++ we typically place constant declarations before any other declarations in the program

Example

• Can you spot what is incorrect in the following program:

```cpp
int main()
{
    const int pi = 3.14;
    double num;
    int i, j;
    num = e2;
    i = 4,000;
    ch = "b"; j = i;
    pi = 5;
    return 0;
}
```

Problem

• The problem specified at the end of class on Wednesday required us to write a program to calculate the area of a circle.

• What constant declarations does our program need?

• What variable declarations does our program need?

Summary

• In today’s lecture we discovered
  o How Data that is used by a program can be declared and stored
  o The difference between constants and variables
  o What constitute valid identifier names
  o The three primitive data types; int, double, char

• We have covered p. 26 - 31 of your textbook