Announcements

- Website is up!
  - http://zeus.pacificu.edu/shereen/CS150
- All lecture slides, assignments, lab notes will be available on this site in two forms:
  - Microsoft PowerPoint or Word
  - PDF (Acrobat Reader)

Implementation

```
//Program purpose:  converts distance in miles to kilometers
//Author:  Friedman & Koffman
//Date:  August 30, 2000
#include <iostream>

int main()
{
        using namespace std;
        const float KM_PER_MILE = 1.609;
        float miles, kms;

        //Get the distance in miles
        cout << "Enter the distance in miles" << endl;
        cin >> miles;

        //Convert the distance to kilometers
        kms = KM_PER_MILE*miles;

        //Display the distance in kilometers
        cout << "The distance in kilometers is" << kms << endl;
}
```

C++ Language Elements

- Comments are
  - how you explain in English what your program does
  - Ignored by the compiler
  - Very, very, very important
- Format of comments:

```
//comment
/* comment */
```

Compiler directives

- `#include <iostream>`
- # signifies compiler directive
- Processed BEFORE program translation
- #include tells the compiler to look for libraries
- <> signifies part of standard C++ libraries
- We'll see other examples later of compiler directives

Namespace std

- `using namespace std;`
- Indicates that we will be using objects that are named in a region called namespace std.
- The statement ends in a semicolon.
- The statement appears in all our programs.

Main function definition

```
int main( )
{
  main program
}
```

- Your main program is where execution starts.
- Every program has one!

Program statements

- Declaration Statements
  - What data is needed by the program?
  - `const float KM_PER_MILE = 1.609;`
  - `float miles, kms;`
- Executable Statements
  - Everything else
  - Examples:
    - `cout, cin`
    - Assignment
- All end with semicolon ;

Identifiers

- Names used in program
- Examples:
  - Variables
  - Functions
- Rules:
  - Begin with letter or underscore
  - Consist of letters, digits and underscore
  - Cannot use reserved word

Identifiers, Contd.

- Reserved Words examples
  - `const, float, int, return, main`
  - Complete list in Appendix B of text
- Case sensitive
- Valid or Invalid?

`Letter1`	`joe's`
`1letter`	`cent_per_inch`
`Inches`	`two-dimensional`
`Inches*num`	`hello`

Data Types and Declarations

- A data type is a way to represent a particular set of values

- Four types
  - Integers
  - Reals
  - Booleans
  - Characters

Integers

- Whole numbers, positive or negative
- Stored as binary number
- Datatype is called int
- Operations?
- Finite
- Examples of integer literals are: 123, -23, 0, 32767

Reals

- Real numbers can contain fractional parts
- Stored in floating point format
- Datatype is float
- Operations?
- Examples of float literals are: 1.0, -.1, 0., 12E5, -1E-2

Characters

- Individual character--letter, digit, symbol
- Characters stored as byte
- Datatype is char
- Operations?
- Char literals are enclosed in single quotes and examples include: 'A' 'a' '?'

Purpose of Datatypes

- Different ones allow compiler to know how to represent value
- Different datatypes can use different operations
- The integer 2 is different from 2.0 and the character 2 (all stored differently)

Declarations

- Declarations are at the beginning of a program
- They list the variables used
- Format:

```
datatype identifier;
```