# Operations involving same and different data types

- Operations involving values from same data type
  - Just like Java:

You can *only* perform operations of 2 values of the *same* data type

• Example:

float A, B, C; - Defines 3 float variables
A = 4;
B = 5;
C = A + B;

This is the similar situation as "adding apples and apples".

### • Operations involving values from *different* data types

- Just like Java:
  - Operations using values of the *different* data type must first convert one type into the other before the operation can be performed.

• Example:

This is the similar situation as "adding apples and oranges". We must convert the apples to oranges first before we can add

#### • Converting between different types of data

- Data conversion must take place in the following operations involving different data types:
  - Binary (arithmethic) operations

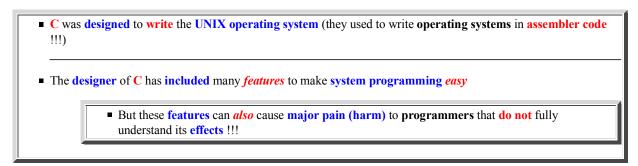
**Example:** 

int a;
float b;

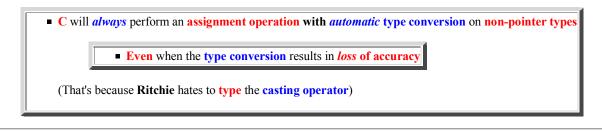
Examj	ple:
	<pre>int a; float b; a = b; // b in converted to a int</pre>
Note:	
	• Yes, this is allowed in C
	In Java, you need to use casting:

#### • Warning: C is for adults *only* !!!

#### • Fact:



#### • Your first feature:

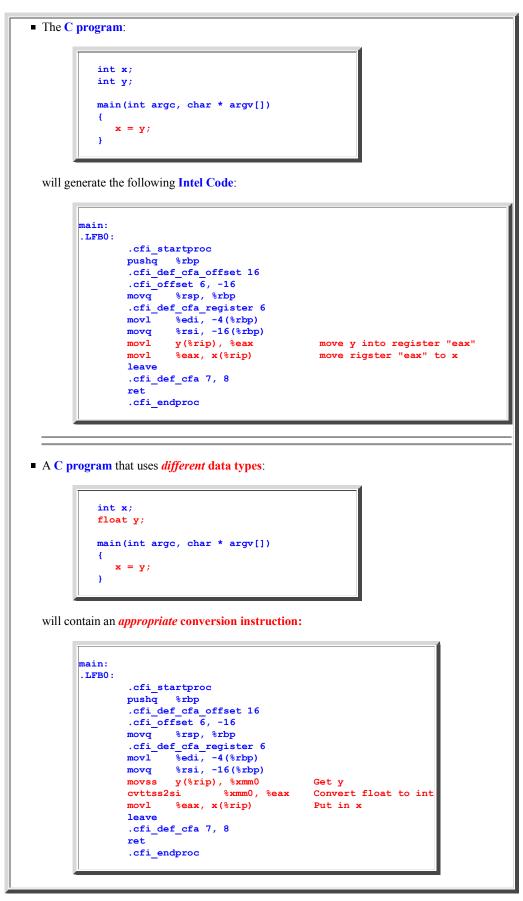


• Making automatic conversion "explicit" with assembler programming

#### • Fact:

• Every computer (CPU) has a number of *conversion* instructions (e.g., int  $\Rightarrow$  float, int  $\Rightarrow$  double and so on)

• We can make the automatic convert by the C compiler explicit by examining the *generated* assembler code:



Here's an explanation of the cvttss2si instruction: click here

# • Example Program: (Demo above code)

# Example

- C program Prog file without the need of conversion: <u>click here</u>
- C program Prog file that uses conversion: <u>click here</u>

## How to run the program:

- Right click on link and save in a scratch directory
- To compile: gcc -S convert?.c
- Examine convert?.s