UNIVERSITY OF MASSACHUSETTS
DARTMOUTH

ECE160: Foundations of Computer Engineering I

Lecture #4 – Data Types and Variables

Instructor: Dr. Liudong Xing
SENG-213C, lxing@umassd.edu
ECE Dept.

Administrative Issues

- Homework #1 assigned
  - Due by **9am, Monday, Jan. 30**
  - Please go to the course website "Homework" section to access the homework #1 problems

    http://xing160.sites.umassd.edu/
  - Please follow the "submission guidelines" to submit your answers (scanned, photo, or typed) to your name folder at the class M: drive

Review of Lecture #3

1. Some definitions and conventions in computer engineering you should know
2. Computer languages evolution: machine → assembly → high-level
3. The first C program
   - preprocessor directives
   - main(), printf()
   - comments
4. A popular software development lifecycle – waterfall model
5. Two types of errors:
   - Syntax: the required form of the program punctuation, keywords (int, float, return, …) etc.
   - Semantics (logic): what the program means or wWhat you want it to do

Temperature Conversion (L#3; Revisit)

```
#include <stdio.h>
int main(void)
{
        float celsius;
        float fahrenheit;
        printf("This program converts Celsius  to Fahrenheit. \n");
        printf("Please enter a Celsius temperature. \n");
        scanf("%f", &celsius);
        fahrenheit = 9.0/5.0 * celsius + 32;
        printf("The temperature in Fahrenheit is: %f\n", fahrenheit);
        return 0;
}
```

Identify the five errors in the program

```
include <stdio.h>
int Main(void)
{
        float celsius;
        float fahrenheit;
        printf("This program converts Celsius  to Fahrenheit. \n");
        printf("Please enter a Celsius temperature. \n");
        scanf("%f", &Celsius);
        fahrenheit = 9.0/5.0 * celsius - 32;
        printf("The temperature in Fahrenheit is: %f\n", fahrenheit)
        return 0;
}
```

Agenda

- Four standard data types in C
  - void, int, char, float
- Variables
  - Declaration and definition
  - Initialization

Textbook: Chapter 2.1, 2.2, 2.4

Types

- A type defines a set of values (*domain of the type*) and a set of operations that can be applied on those values

- Example: a light switch (compared to a computer type)
  - Its domain consists of two values: on (1), off (0)
  - Only two operations can be applied to it: turn-on, turn-off

Types in Computers

- Data have types
- Functions also have types. This is the type of the data they return.

Data Types

- Standard types
  - void
  - int: integer
  - char: character
  - float: floating point
- Derived types
  - Complex structure built using standard types
  - E.g.: pointer, array
  - To be discussed later

void

- Has no values
- Has only one operation: assignment

Integer (int)

- A number without fraction part
- C has 3 different sizes of the int type:
  - short int
  - int
  - long int
- The size of int is machine dependent !!!!
- To find the size on your machine use the operator *sizeof*

Signed and Unsigned Integers

- If the integer is signed, then one bit must be used for the sign (0 is plus, 1 is minus).

- As a result, the maximum value of an unsigned integer is twice as large as the maximum value of a signed integer.

Logical Data (Boolean)

- Can only be true or false.
- C supports logical data type through the integer type
  - any nonzero number is considered true.
  - zero is considered false.

Character (char)

- A character is a value that can be represented in the computer's alphabet.
- Most computers use 1 byte to represent characters.
  - ASCII: American Standard Code for Information Interchange (https://www.ascii-code.com/)
  - e.g.: letter a is a binary 01100001 (61 Hex; 141 Oct; 97 Dec)
- A character in C can be interpreted as a small integer (0 ~ 255). For this reason, C often treats a character like an integer

Floating Point (float)

- 3 types of floats:

  float, double, long double

Type	Byte size
float	4
double	8
long double	10

- Note: floats are always signed.

Agenda

- ✓ Four standard data types in C
  - void, int, char, float
- Variables
  - Declaration and definition
  - Initialization

Textbook: Chapter 2.1, 2.2, 2.4

Variables

- Variables are named memory locations that have a
  - Type (e.g., int, and consequently a size)
  - Identifier (name; follow rules in L#3 Slides 22, 23),
  - Value.

Exercises (1)

- Which of the following are incorrect variable names and why?

  Cat

  A+B123

  Bα3

  2dogs

  lotus12

Variable Declaration and Definition

- Each variable in the program must be declared and defined!
  - Declaration: to name a variable
  - Definition: to create a variable, to reserve memory for it
  - Usually, a variable is declared and defined at the same time!

Variable Declaration and Definition

- Examples

```
float temperature;
int age;
float payRate;
long int national_debt;
double tax;
char code, kind;
```

Note the ; after each declaration

- C allows multiple variables of the same type to be defined in one statement

Exercises (2)

- *True or false:* the following two statements are identical

```
int abc, DEF;
int ABC, def;
```

Variable Declaration and Definition

- Good programming practice
  - Have one declaration per line
  - Variable identifiers: use **one** style that make them readable

Variable Initialization

- When a variable is defined, it is not initialized automatically!
- The programmer must initialize any variable requiring prescribed data when the function starts
- Use assignment operator

  variable_name = value;

- Example:

```
temperature = 78;
age = 18;
tax = 730;
```

Variable Initialization

- C allows a variable to be defined and initialized at the same time
- Examples

```
int age = 18;
float temperature = 78;

int count, sum=0; /*Only sum is initialized!*/

int count=0, sum=0;
/*Both count and sum are initialized! But we prefer:*/

int count=0;
int sum=0;
```

Variable Initialization

- Remember to initialize variables

- Good programming practice
  - One initialization per line
  - Separate variable declaration from variable initialization

Exercises (3)

- Which of the following are incorrect C assignment statements and why?

  Year = 1975

  1973 = oldyear;

  Day = 24 hours;

  Age = 32;

Exercises (4)

- Declare and define two variables (num1, num2) of integer type, a variable (float1) of double floating point, and a variable (chara1) of character type. And initialize them to be 0, 0, 0, ‘a’, respectively.

Summary of Lecture#4

- Four standard data types in C: void, int, char, float
- Each variable in the program must be declared and defined!
- C allows multiple variables of the same type to be defined in one statement
- When a variable is defined, it is not initialized automatically!
- The programmer must initialize any variable requiring prescribed data when the function starts

Textbook: Chapter 2.1, 2.2, 2.4

Things To Do

- Homework #1 assigned
  - Due by **9am, Monday, Jan. 30**
  - Please follow the "submission guidelines" on the course website to submit your answers to your name folder at the class M: drive

- Review Lab1 assignments

Next Topic: Constants