CS50 Section 0: The Basics of C and CS

Ashin Shah (ashah@fas)

09.19.2010

About Me

```
Typedef struct node {
  char[] name = "Ashin";
  char[] concentration = "Applied Math";
  int year = 3;
  char[] house = "Pfoho";
}TF;
```

Section Overview

4:37 Review of Last Week's Lectures
5:00 Extensions on Material
5:30 Questions/Week Ahead

NO laptops in Section.
Ironic, I know.

Course Expectations

Weekly Problem Sets, Quiz 0, Quiz 1, Final Project

Resources

Lecture, Videos, Scribe Notes, Section, Section Notes, CS50 Support, ***Walkthroughs***, TF/CA, OH, VOH, cs50.net, Readings...

Review of Week 0, Week 1, Week 2

What is Computer Science? What is programming?

Algorithms: Binary Conversion / Why do we even care? _____

2^9	**2^8**	**2^7**	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**

Construct an algorithm that will convert a number from Base 10 to Base 2.

Homework: *Given a Base 2 value, compute its Base 10 representation. Now extend your algorithm to convert any Base A value to Base B. Assume* $A,B \in [2,10]$.

C Syntax

Libraries – Remember **#include <***[library_name].h***>**
Don't forget the ; after each line
Every { needs a }, every (needs a)
INDENT

```
/********************
 * STUFF
 ********************/

#include <cs50.h>

int
main(void)
{
    CODE GOES HERE;
}
```

How to Make a Program (Flowchart)

Getting Machines to Do Your Bidding

Primitive Types
- Variables

```
int val = 3;
int val2 = val * 2;
printf("%d", val2);
```

Operator Precedence / Logic
- Arithmetic: +, -, *, /, %
- Relational: ==, !=, >, <, >=, <=
- Logical: !, &&, ||
- Assignment: =, +=, -=, *=, /=, %=
- Decrement/Increment: ++, --
- Casts

De Morgan's Laws

```
!(A && B)  ==  !A || !B
!(A || B)  ==  !A && !B
```

Control Structures: If/Else == Switch

Datatypes (Bytes), Range:
Int (4), -2^31 to 2^31-1
Char (1), 0 to 255, or -128 to 127
Float (4),
Double (8),
Long (4),
Long Long (8), -2^63 to 2^63-1
Short (2), -2^15 to 2^15 - 1
Unsigned int (4), 0 to 2^32-1

```
If(condition) {DO THIS;}
Else if(condition2) {DO THIS;}
Else if(condition3) {DO THIS;}
…
Else {THIS IS DONE;}
```

Loops = {"For", "While", "Do-While"}

- *Initialization*
- *Condition*
- *Update*

For	While	Do-While

Nesting

User Input
- #include <cs50.h>
- Get*[type]*()

User Output
- printf("TEXT TO BE DISPLAYED",,,…);
  - Formatting codes

This is CS50 [Section]. Welcome.

<Section Notes Compiled by Ashin Shah>

VL50 LXVMBHG 1: VKRIMHZKTIAR

Ashin Shah (ashah@fas)

09.26.2010

BinaryConvertAlgorithm

Grading
Correctness, Style, Design

Questions?

Section Agenda
4:37 Functions, Variable Scope
5:00 Arrays
5:25 Cryptography
5:50 Questions/Week Ahead

Functions

```
double sin_times_cos(double x, double y) {
    return sin(x) * cos(y);
}
```

3 Purposes of Functions

1.

2.

3.

ANATOMY OF A FUNCTION

```
double accumulate_interest(double balance, double rate);

...

double accumulate_interest(double balance, double rate) {

   double accrued; double updated;

   accrued = balance * rate; updated = balance + accrued;

   return updated;

}
```

Parts of a Function

(A) RETURN
(B) DEFINITION
(C) FUNCTION NAME
(D) LOCAL VARIABLES
(E) RETURN TYPE
(F) PARAMETER LIST
(G) DECLARATION
(H) FUNCTION CALL
(I) BODY
(J) HEADER

ELSEWHERE, WITHIN THE MAIN METHOD

```
double money = accumulate_interest(3000.0, 1.2);
```

More on Variables

Variable Scope: LOCAL vs. GLOBAL?

M.A.G.I.C. – DON'T DO IT
#DEFINE constants

```
int
main(int argc, char *argv[]) {

    string tf_array[30];
    . . .

    // populate the array

    // print out the contents of the array
    for (i = 0; i < 30; i++) {
        printf("%s\n", tf_array[i]);
    }
}
```

Arrays

```
int votes[3];

votes[0] = 10;
// stuffing the ballot box
votes[1] = 0;
votes[2] = 0;
```

Some key properties of C arrays:

- Zero-indexed
- Access beyond bounds
- Array names are not variables

Array Initialization

```
int num_days_in_month[12] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
string four_elements[] = { "fire", "water", "wind", "earth" };
bool truth_table[2][3] = { { true, true, true }, { true, true, true } };
```

```
#define CLASS_SIZE    30
#define EMPTY    0
void init_array(int array[], int size);
```

```
int
main() {
     int i;
     int scores_array[CLASS_SIZE];

     init_array(scores_array, CLASS_SIZE);

     // populate the array
     for (i = 0; i < CLASS_SIZE; i++) {
          printf("Enter score for student %d: ", i);
          scores_array[i] = GetInt();
     }
}

void init_array(int array[], int size) {
     int i;

     // initialize all cells to 0
     for (i = 0; i < size; i++) {
          array[i] = EMPTY;
     }
}
```

Cryptography

What is cryptography?

Caesar Ciphers

$$c_i = (p_i + k) \bmod 26$$

The Vigenere Cipher

$$c_i = (p_i + k_i) \bmod 26$$