This is CS50 (section).

Tommy MacWilliam, 2010

Announcements

- Problem Set 2 walkthrough
- Pset0 feedback: 9.24.10
- Pset1 feedback: 9.27.10
- Office Hours
- help.cs50.net

Today

- Functions
- Scope
- Magic
- Arrays
- Main
- Chars and casting
- Cryptography

Functions

- Group of code to accomplish a specific goal
- Takes input, generates output

Functions

- Reusable—DRY up your code
  - *D*on't *R*epeat *Y*ourself
- Break up large code chunks
  - Make a big problem into a series of smaller problems
- Organize your code

Functions

```
<type>
<name> (<arguments>)
{
    <code>
}
```

Functions

- Type: int, void, float, etc.
- Name: contains letters, numbers, and underscores
  - Start function names with a verb
- Arguments: input

Functions

- Example time!

Variable Scope

- Global variables
  - Defined outside of all functions
  - Accessible by all functions
- Local variables
  - Defined within a function, only accessible in that function

Variable Scope

- If a global variable and local variable have the same name, the local variable will be used
- Any function can modify a global variable (and thus change the behavior of others)
- This is where things can get confusing

Variable Scope

- Example time!

Magic Numbers

- Constants within your program

```
for (int i = 0;
     i < 4;
     i++) { // code }
```

Magic Numbers

- Bad plan
- `#define` is your friend
  - `#define CONSTANT 4`
- Give meaning to your constants
- Easily changeable if used throughout your program

Magic Numbers

- Constants created using `#define` are NOT variables
- The compiler literally replaces `CONSTANT` with `4`

Arrays

- List of elements of the same type
- Elements accessed using an index (aka position) in the array
  - Index starts at 0!
- `int array[3] = {1, 2, 3};`
- `array[0] = 1;`

Arrays

- Example time!

Arrays

- Can also be multi-dimensional
- Use multiple indexes
  - `int grid[3][5]`
  - The “grid” array will have 3 rows and 5 columns

Arrays

- Example time!

Main

- Main is a *function* that takes two arguments
  - argc: number of arguments given to the program
  - argv: *array* of arguments

Chars and Casting

- Every char has an associated int
  - A = 65, a = 97, etc.
- Use casting to “convert”
  - `int n = (int) 'c';`
  - `char l = (char) 84;`

Chars and Casting

- Example time!

Cryptography

- Review of terms:
  - Cleartext: the message you want to encrypt
  - Ciphertext: your message after you encrypt it
  - Key: what you use to encrypt the message

Caesar

- Rotate each character by single character key
  - Each character rotated by the same amount

Caesar

- Key = 5, message = “hello”
  - H + 5 = M
  - E + 5 = J
  - L + 5 = Q
  - O + 5 = T
- Ciphertext = “mjqqt”

Vigenère

- Rotate each character according to a multi-character key
  - Each character not rotated by the same amount

Vigenère

- Key = “not”, message = “here”
  - H + N = U
  - E + O = S
  - R + T = K
  - E + N = R
- Ciphertext = “uskr”