CSC209 Lecture 5: Files

David Liu, Department of Computer Science, University of Toronto

*Navigation tip for web slides: press **?** to see keyboard navigation controls.*

Announcements

- Assignment 2 has been posted!

Files: five standard actions

Action	C function(s)
Open a file	`FILE *fopen(const char *pathname, const char *mode)`

Read from a file	`// Text I/O`
	`int fscanf(FILE *stream, const char *format, ...)`
	`char *fgets(char *s, int size, FILE *stream)`
	
	`// Binary I/O`
	`size_t fread(void *ptr, size_t size, size_t nmemb, FILE *stream)`

Write to a file	`// Text I/O`
	`int fprintf(FILE *stream, const char *format, ...)`
	
	`// Binary I/O`
	`size_t fwrite(const void *ptr, size_t size, size_t nmemb, FILE *stream)`

Change position in a file	`int fseek(FILE *stream, long offset, int whence)`
Close a file	`int fclose(FILE *stream)`

“Text” vs. “binary” files

message.txt

```
David is cool
```

armin.jpg

Demo with od:

```
$ od --address-radix=x --format=c --format=dC <FILE>
```

Worksheet: io_conversions.pdf

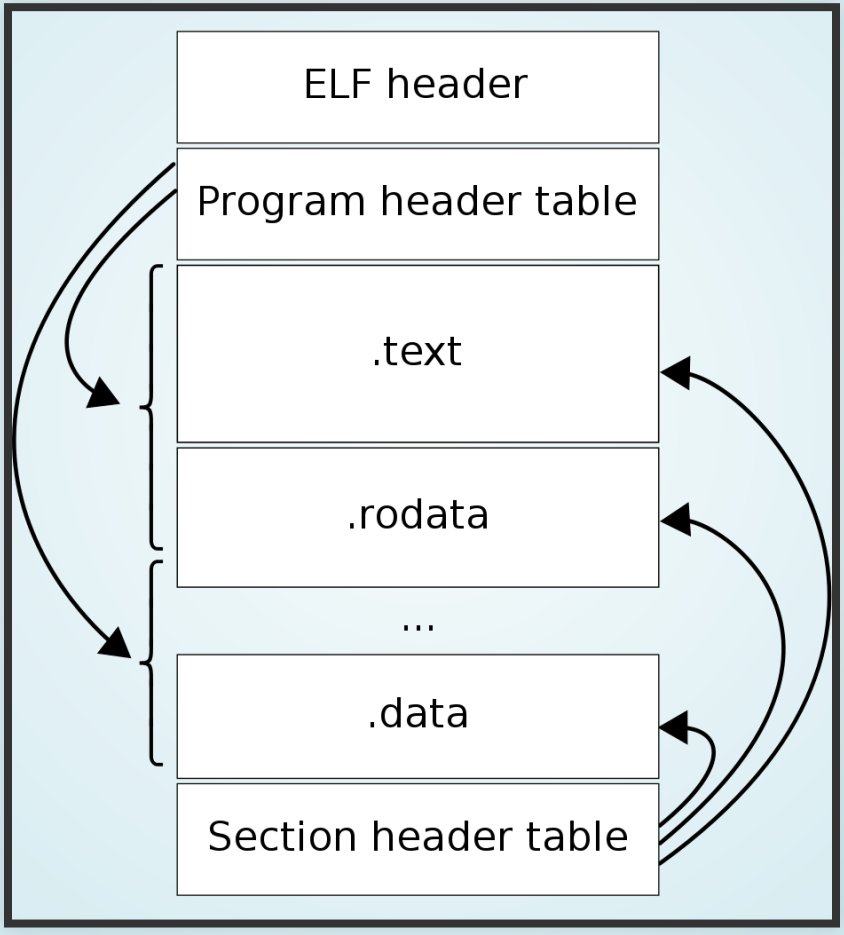
Executables as binary data

```
$ gcc -Wall -std=gnu99 -g -o hello hello.c
```

`hello` is an **executable file**—its contents are binary, not text, data!

The file format is called the **Executable and Linkable Format (ELF)**.

(Don't worry about the details on this slide!)



Inspecting the executable using `objdump`

`objdump` is a utility program for reading object and executable files.

Reading all section metadata:

```
$ objdump --wide --sections <FILE>
```

Reading section metadata for a specific section:

```
$ objdump --wide --sections --section=.rodata <FILE>
```

Reading the contents of a section metadata:

```
$ objdump --full-contents --section=.rodata <FILE>
```

Worksheet: `literals.pdf`

Starter code:

- Part 1: `literals.c`
- Part 2: `rodata.c`