

C Programming in Linux

Aditya Godbole
Cameo Digital Systems

MIT COE 2009

Fundamentals

Fundamentals

Let us C ...

What is C?

Line 1 Col

/*

GRAPHICS DEMO FOR

Copyright (c) 1987

From the command 1

Compile to OBJ C:BGIDEMO.OBJ

Make EXE file C:BGIDEMO.EXE

Link EXE file

Build all

Primary C file:

Get info

C:BGIDEMO.C

ts reserved.

tcc bgidemo graphics.lib

*/

#ifdef __TINY__

#error BGIDEMO will not run in the tiny model.

#endif

#include <dos.h>

#include <math.h>

#include <conio.h>

Message

“You used VC? I did it using C.”

Language

C, C++, Java, Lisp, Perl, Ruby

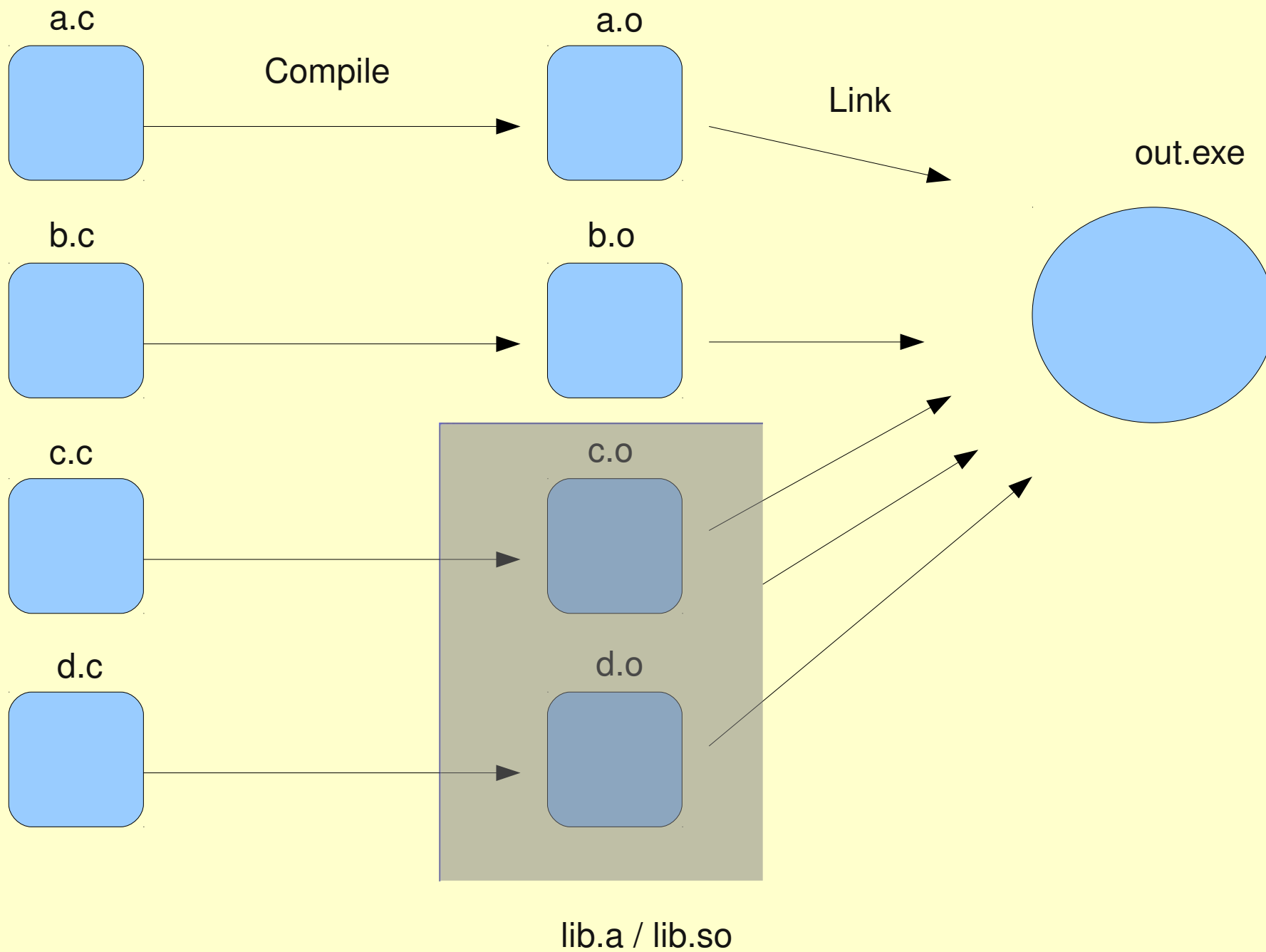
Compilers / Interpreters

Convert code that humans read and write to code that computers can operate on.

Turbo C, VC, GCC, Intel C compiler, etc.

Executables

- Code that the computer can understand
- Headers that the operating system can understand
- Made up of many parts linked to each other



Libraries

- Advantages -
 - Reusable
 - Code need not be compiled every time with the rest of the files
 - Source code not necessary to use the functionality
- Standard C library - libc
 - Functionality that is part of the ANSI C standard.
 - libx11, libqt, libcv, libblas, libfann1

Headers

- C is strongly typed
- Prototype required to check function calls
- Prototypes (declarations) of functions stored in header libraries.
- Necessary to compile function calls to libraries.

GCC

- `gcc -o out -lm a.c b.c c.c`
- `-o`: Name of output executable
- `-l`: Name of libraries to be linked

Migration tips

- `conio.h`
 - Used typically for a `getch`
 - Forget it, not needed
 - Use `getchar` instead
- `graphics.h`
 - X11, Qt
 - Graphics compatible library (incomplete). You can add on
- `math.h` functions
 - Compile with `-lm`

- Use 'sizeof'. Don't assume that an integer is 16 bits or 32 bits
- Read and assign members of data structures individually
- fflush(stdin) does not work! Read the documentation of the standard library in detail
- Write modular code. Break your code into multiple files if required
- Use a **good text editor** for writing programs.

Help !!

- Manual pages – Detailed information about every command and even about libraries
- Google (or your favourite search engine)
- Specific mailing lists (or google groups). Follow the etiquettes of the group.
- Learn to ask. The best learners are those who ask the best questions.
- Share your knowledge. Its an integral part of getting help.

Must reads

- The C Programming Language, Brian Kernighan and Dennis Ritchie
- The Art of Unix programming, Eric Raymond
- Let Us C, Yashwant Kanitkar

Must throw away

- Let Us C, Yashwant Kanitkar

Thank You

Questions?