Typical C/C++ Language Processing

- **file.c**
  - or
  - **file.cpp**

  **cpp**
  (preprocessor)

  **file.i**

  **cc1**
  (C compiler)

  **file.s**

To do all the steps, type:
- gcc file.c (for C, or)
- g++ file.c (for C++)

Handles \\#include, \\#define, \\#if, etc.

To Stop here, type:
- g++ -E file.cpp

This is the actual compiler

To stop here, type:
- g++ -S file.cpp
C/C++ Language Processing (cont’d)

Translates from assembly language into machine (binary) - unique to each processor type

Object file - to stop here:
```
g++ -c file.cpp
```
It is common to stop here

"Knits" together (links) all of the independent program modules

This is probably called
```
file.exe
```
on Windows systems
ELF - Relocatable Object File

Sections

- .text
- .rodata
- .data
- .bss
- .symtab
- .rel.text
- .rel.data
- .debug
- .line
- .strtab

Describes object file sections

Section Header Table

ELF Header
# ELF – Executable File

<table>
<thead>
<tr>
<th>ELF Header</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment Header Table</td>
</tr>
<tr>
<td>.init</td>
</tr>
<tr>
<td>.text</td>
</tr>
<tr>
<td>.rodata</td>
</tr>
<tr>
<td>.data</td>
</tr>
<tr>
<td>.bss</td>
</tr>
<tr>
<td>.symtab</td>
</tr>
<tr>
<td>.debug</td>
</tr>
<tr>
<td>.line</td>
</tr>
<tr>
<td>.strtab</td>
</tr>
</tbody>
</table>

**Section Header Table**

- **Read-only memory segment (code)**: .init, .text, .rodata
- **Read-Write memory segment (data)**: .data, .bss
- **Symbol table & debugging info (Not loaded into memory)**: .symtab, .debug, .line, .strtab
LINUX Runtime Memory Image

Kernel virtual memory

User stack

Shared Libraries

Run-time heap

Read-Write segment (.data, .bss)

Read-only segment (.init, .text, .rodata)

Unused

0xc0000000

0x40000000

0x00400000

0x00400400

%esp

brk

Loaded from executable file

Inaccessible to user code
/* crt1.o */

_start:
    call __libc_init_first /* entry point in .text */
    call _init /* startup code in .text */
    call atexit /* startup code in .init */
    /* set up argument list for main here */
    call main /* startup code in .text */
    call _exit /* application main code */
    /* returns control to shell */