CIS 190: C/C++ Programming

Lecture 3

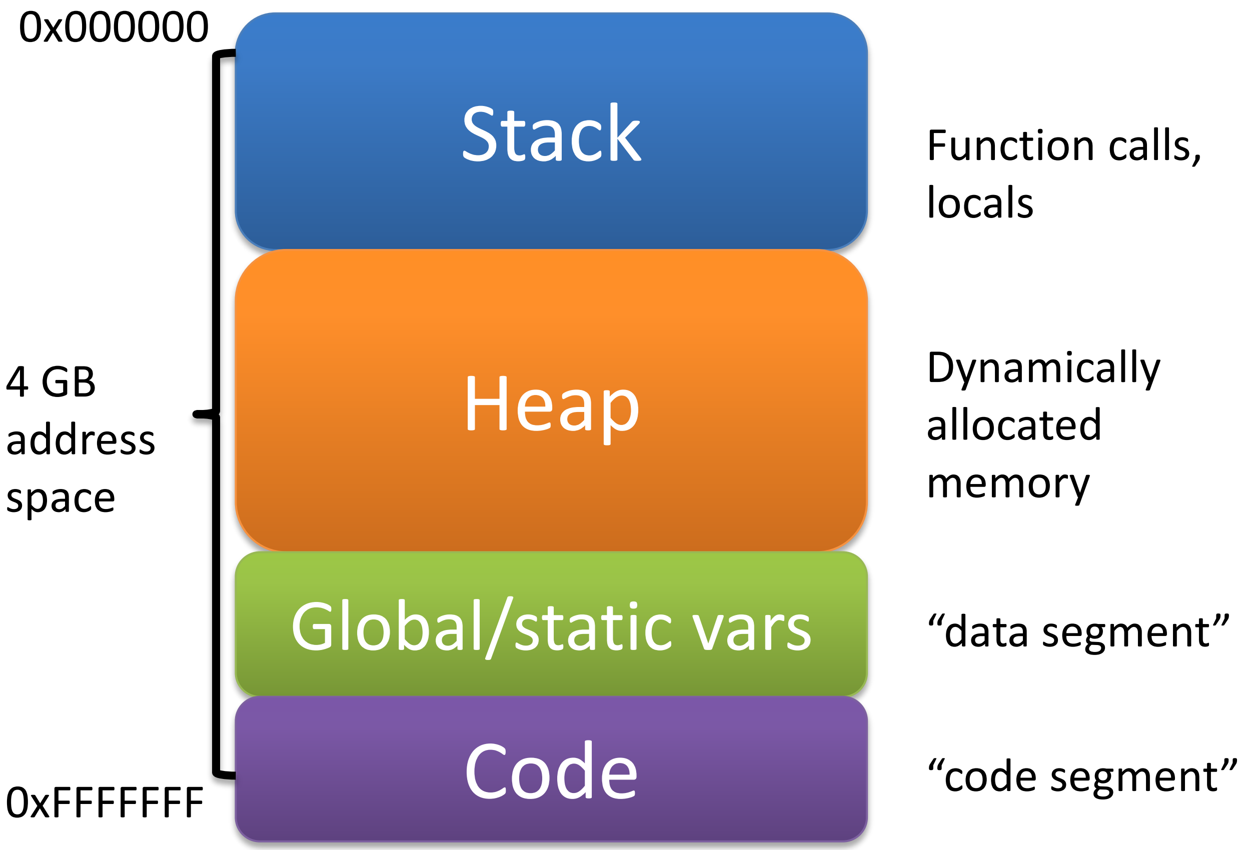
Memory Management in C

Outline

- Memory allocation
- Memory errors
- Debugging
- Makefiles

Memory

- each process gets its own memory chunk, or *address space*



0x000000

4 GB address space

0xFFFFFFF

Stack

Function calls, locals

Heap

Dynamically allocated memory

Global/static vars

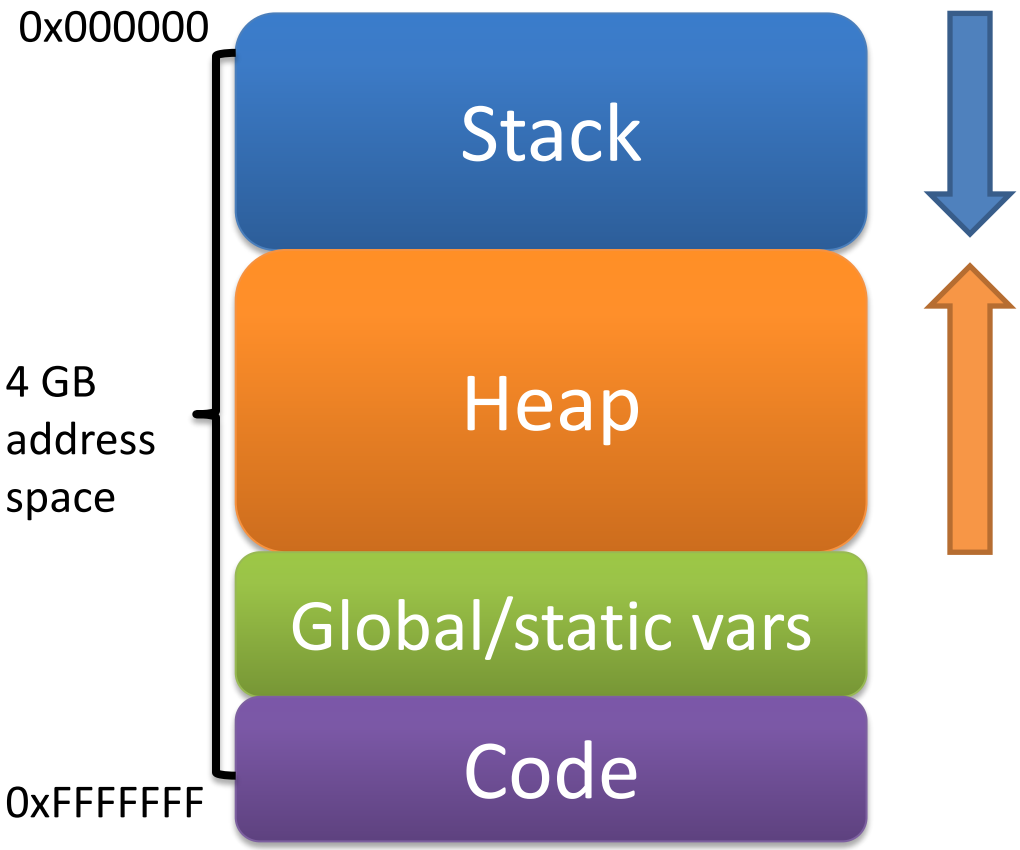
“data segment”

Code

“code segment”

Memory

- each process gets its own memory chunk, or *address space*



Stack Allocation

- memory allocated by the program as it runs
  - local variables
  - function calls

- fixed at compile time
  - can’t be changed while running

Heap allocation

- dynamic memory allocation
  - memory allocated at run-time

- two options for allocating memory:
  - **`malloc()`**
  - **`calloc()`**
    - both require **`#include <stdlib.h>`** to work

malloc()

```
<type>* malloc ( <size to be allocated> )
```

- malloc returns a pointer of the size requested

```
char *letters;
letters = (char*) malloc(userNumLetters *
                                            sizeof(char));
```

- cast malloc to be a pointer of desired type
- call with how many elements needed, multiplied by the size of that element type

calloc()

```
<type>* calloc (<number of elements>,
                <size of each element> )
```

- calloc works very similarly to malloc, but it initializes all the allocated bits to zero

```
float *grades;
grades = (float*) calloc(userNumStudent,
                         sizeof(float));
```

- calloc takes longer than malloc

Correctly handling memory

- IMPORTANT: before using allocated memory make sure it’s *actually been allocated*

- if memory wasn’t correctly allocated, the pointer will be null, and your program should exit

```
if (grades == NULL)
{
  printf(“Memory not allocated, exiting.\n”);
  exit(-1);
}
```

Outline

- Memory allocation
- Memory errors
- Debugging
- Makefiles

Segmentation faults

- when program tries to access a memory location forbidden by the OS

- common causes for seg faults:
  - accessing out-of-bounds on an array
  - uninitialized pointers

Freeing memory

- stack allocated memory is automatically freed whenever functions **`return`**

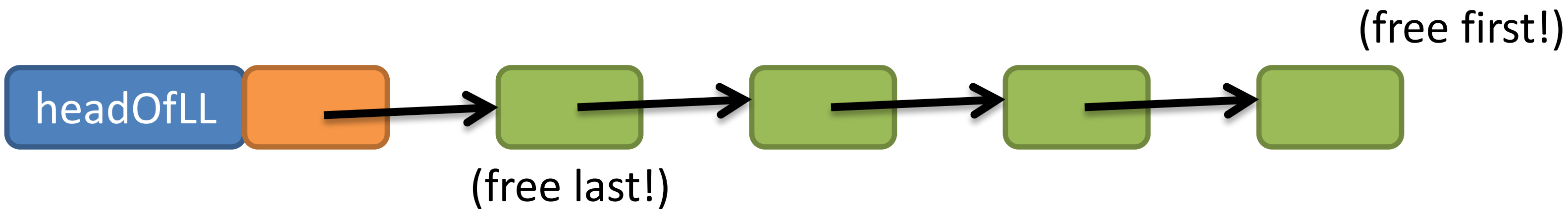
- heap allocated memory was allocated by you – so it must also be freed by you
- done using the **`free()`** function

```
free( <name of pointer to memory> )
```

Freeing in order

- `free()` does not work recursively
- need to free each block of memory separately

- free must be called in a sensible order



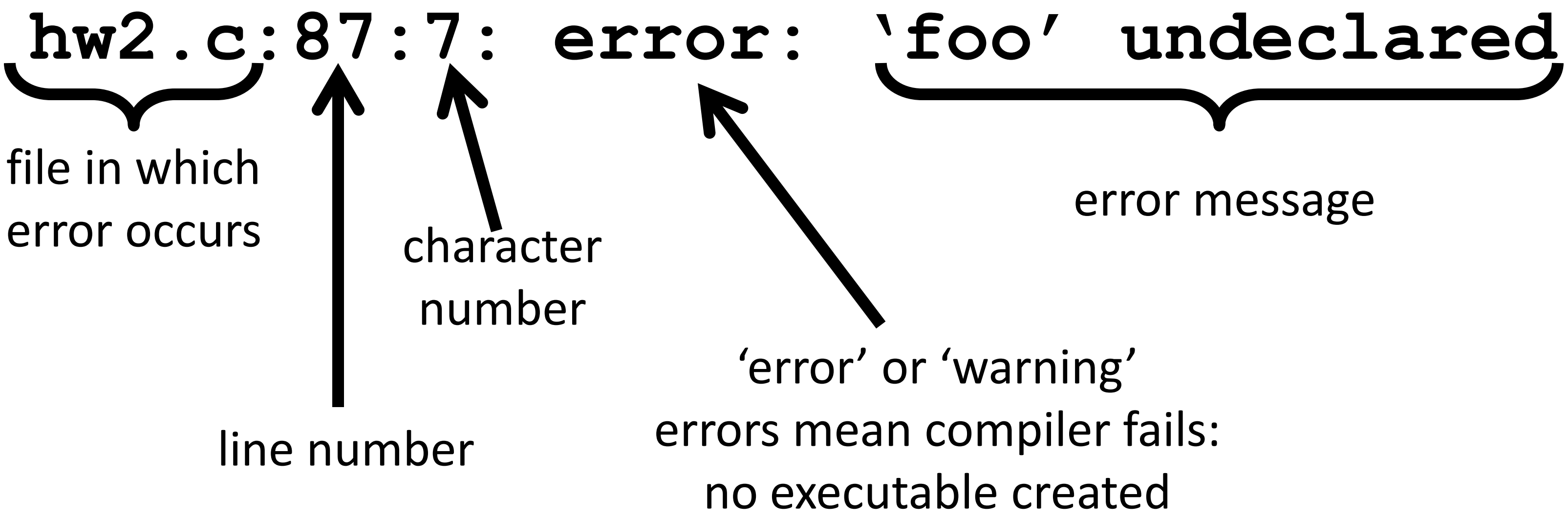
Memory mistakes

- memory leaks
  - when data is allocated but not freed
  - access to memory is lost, example:
    - a loop that re-allocates memory to the same variable
- double `free()`ing
  - freeing a pointer twice
- dangling pointer
  - a pointer that points to memory that was freed

Outline

- Memory allocation
- Memory errors
- Debugging
- Makefiles

Understanding Errors



```
hw2.c:87:7: error: ‘foo’ undeclared
```

file in which error occurs

line number

character number

‘error’ or ‘warning’
errors mean compiler fails:
no executable created

error message

- look at the line of code (then the one before)
  - shortcut to go-to line in xemacs:
    ESC+g, then line number, enter

Debugging Basics

- start with the **very first** error and recompile every time an error is fixed
  - errors will cascade (and de-cascade when fixed)
- probe statements with printf
  - adding a printf might change your error!
- GOOGLE

Common Compiler Errors part 1

```
hw2.c:87:7: error: ‘foo’ undeclared
```

- for variables: forgot to declare, or misspelled
- for functions: misspelled function, or forgot to #include the file containing the prototype

```
hw2.c:37:6: warning: unused variable
   ‘bar’
```

- variable was declared but not used

Common Compiler Errors part 2

```
hw2.c:54: warning: suggest
   parentheses around assignment
   used as truth value
```

- usually a mistake: you meant to use **==** not **=**

```
hw2.c: 51: error: expected ‘;’
   before ‘for’
```

- missing semicolon on previous line of code

Common Linker Errors

```
hw4.o: In function ‘main’:
hw4.c:91: undefined reference to ‘Fxn’
```

- linker can’t find code for ‘Fxn’ in any .o file
  - forgot to link .o file
  - misspelled named of Fxn
  - parameter list is different

```
/usr/lib64/gcc/[...]/crt1.o: In function
   ‘_start’:
/home/[...]/start.S:119: undefined reference
   to main
```

  - you compiled a file that does not contain main() without using the –c switch to indicate separate compilation

Easy error to fix

```
> gcc -Wall structs.c
In file included from /usr/include/stdio.h:33:0,
                 from structs.c:6:
/usr/lib64/gcc/x86_64-suse-linux/4.7/include/stddef.h:213:1: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'typedef'
In file included from /usr/include/stdio.h:74:0,
                 from structs.c:6:
/usr/include/libio.h:307:3: error: unknown type name 'size_t'
/usr/include/libio.h:311:67: error: 'size_t' undeclared here (not in a function)
/usr/include/libio.h:339:62: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/libio.h:348:6: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/libio.h:470:19: error: expected '=', ',', ';', 'asm' or '__attribute__' before '_IO_sgetn'
In file included from structs.c:6:0:
/usr/include/stdio.h:319:35: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:325:47: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:337:20: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:344:10: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:386:44: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:390:45: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:666:11: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:669:9: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:679:8: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdio.h:709:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'fread'
/usr/include/stdio.h:715:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'fwrite'
/usr/include/stdio.h:737:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'fread_unlocked'
/usr/include/stdio.h:739:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'fwrite_unlocked'
In file included from structs.c:9:0:
/usr/include/string.h:43:8: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:46:56: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:55:18: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:62:42: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:65:56: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:92:48: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:129:39: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:137:9: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:143:57: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:150:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'strxfrm'
In file included from structs.c:9:0:
/usr/include/string.h:165:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'strxfrm_l'
/usr/include/string.h:180:45: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:281:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'strcspn'
/usr/include/string.h:285:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'strspn'
/usr/include/string.h:395:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'strlen'
/usr/include/string.h:402:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'strnlen'
/usr/include/string.h:423:12: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:447:33: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:451:53: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:455:31: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:458:54: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:536:61: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:573:34: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/string.h:576:39: error: expected declaration specifiers or '...' before 'size_t'
In file included from structs.c:11:0:
/usr/include/stdlib.h:139:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before '__ctype_get_mb_cur_max'
In file included from structs.c:11:0:
/usr/include/stdlib.h:331:4: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:361:4: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:465:22: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:467:22: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:467:38: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:479:36: error: expected declaration specifiers or '...' before 'size_t'
In file included from /usr/include/stdlib.h:491:0,
                 from structs.c:11:
/usr/include/alloca.h:32:22: error: expected declaration specifiers or '...' before 'size_t'
In file included from structs.c:11:0:
/usr/include/stdlib.h:497:22: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:502:45: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:502:65: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:755:9: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:755:25: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:760:34: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:760:50: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:839:6: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:842:6: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:846:31: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:850:31: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:859:36: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:863:34: error: expected declaration specifiers or '...' before 'size_t'
/usr/include/stdlib.h:870:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'mbstowcs'
/usr/include/stdlib.h:873:15: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'wcstombs'
```

Debuggers

- see what is going on “inside” the program
  - more powerful and accurate than printf() probes

- examine variables (value and address)
  - change variables on the fly

- step through code line by line
  - skip blocks of code you don’t want to see

Using DDD (or GDB)

- add the ‘**−g**’ flag when compiling
- open program for testing using ddd [executable] or gdb [executable]
- add breakpoints to stop the program at specific points
- use ‘print’ or ‘display’ to show values (or addresses) of variables
- step through code line by line

DDD Tips

- File -> Open Source
  - choose a different file to look at (and to set breakpoints in)
- Source -> Reload Source
  - refresh the source you're using after recompiling without losing any breakpoints or data displays
- FINISH
  - executes the current "frame"
  - will pause when it hits a return (outside of main)

DDD Livecoding

- DDD livecoding example was taken wholesale from the sample session on this page:
  - http://www.gnu.org/software/ddd/manual/html_mono/ddd.html

- more information about using DDD

- GDB is the non-graphical version of DDD

Outline

- Memory allocation
- Memory errors
- Debugging
- Makefiles

Makefile

- required for all future homeworks
  - makes it easier to grade
  - makes it more accurate to grade

- **#1 rule**: "`make`" fully compiles program
- needs to be called Makefile

Homework 2

- due tomorrow night @ midnight

- clarifications made on the homework page
  - no warnings when compiling
  - turnin instructions

- if you haven’t started yet – do it NOW!

Homework 3

- Memory Diagrams

- write legibly
- double check your work

- due at BEGINNING of class, on paper
  - no late days for this homework!