

Finale

Last Day! Seg Fault

CS 2130: Computer Systems and Organization 1

December 5, 2022

Announcements

- Homework 8-10 due tomorrow at 11pm
 - Homework 10 is **optional** - replaces lowest homework grade
- Final Exam: December 15 at 7-10pm
 - Review session next week (more details soon!)

Announcements

- Homework 8-10 due tomorrow at 11pm
 - Homework 10 is **optional** - replaces lowest homework grade
 - Late submission for all allowed through Friday (no penalty)
- Final Exam: December 15 at 7-10pm
 - Review session next week (more details soon!)

Announcements

- Homework 8-10 due tomorrow at 11pm
 - Homework 10 is **optional** - replaces lowest homework grade
 - Late submission for all allowed through Friday (no penalty)
- Final Exam: December 15 at 7-10pm
 - Review session next week (more details soon!)
- Please complete student evaluations **by Friday 12/9**

Announcements

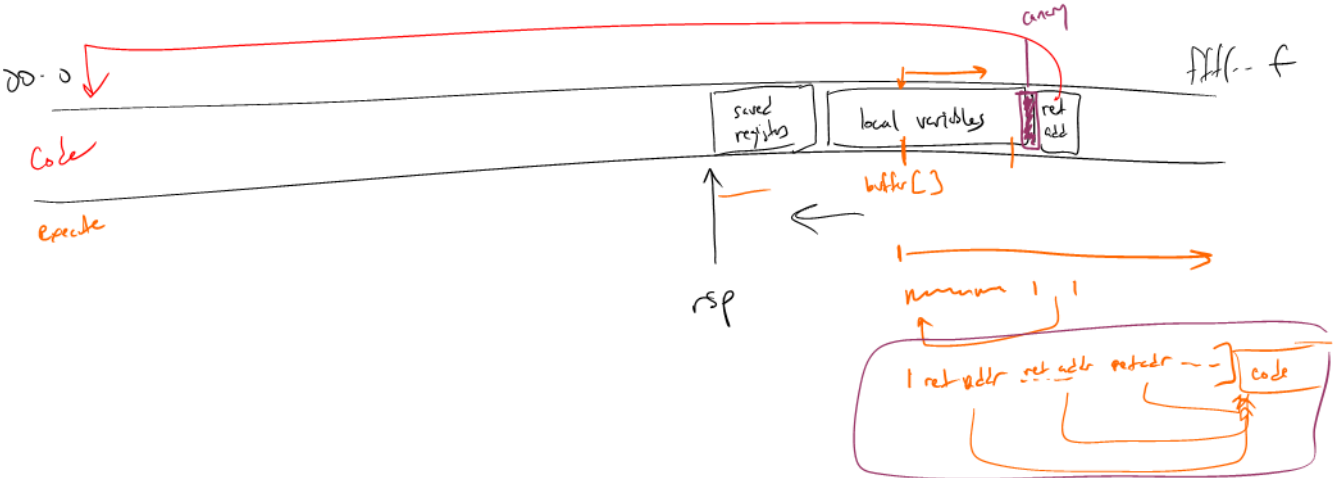
- Homework 8-10 due tomorrow at 11pm
 - Homework 10 is **optional** - replaces lowest homework grade
 - Late submission for all allowed through Friday (no penalty)
- Final Exam: December 15 at 7-10pm
 - Review session next week (more details soon!)
- Please complete student evaluations **by Friday 12/9**
 - 5pts Extra Credit on Final Exam

Pig Latin example

Common Memory Problems (from reading)

- Memory leak
- Uninitialized memory
- Accidental cast-to-pointer
- Wrong use of 'sizeof'
- Unary operator precedence mistakes
- Use after free
- Stack buffer overflow
- Heap buffer overflow
- Global buffer overflow
- Use after return
- Uninitialized pointer
- Use after scope

Vulnerabilities



Vulnerabilities

Anytime you can modify memory the programmer did not expect you to be able to modify, there's something you can do to give yourself power or rights the programmer didn't mean to give you

Vulnerabilities

Vulnerability: a program for which something like this could happen (security holes)

- Ex: stack buffer overflow possibility
- Not necessarily malicious (like when we talked about backdoors)

Exploit: a way to use a vulnerability or backdoor that has been created

- Ex: the magic long word to type into our program

Vulnerabilities

What should you do when you find a vulnerability?

Good Practices

Good practices when finding a vulnerability:

1. Tell the owner
2. Wait (a reasonable amount of time for a fix)
3. Publish

Finale

We have covered a LOT

Finale

We have covered a LOT

- Electricity on wires

Finale

We have covered a LOT

- Electricity on wires
- Transistors to gates (AND, OR, ...)

Finale

We have covered a LOT

- Electricity on wires
- Transistors to gates (AND, OR, ...)
- Combined gates to make circuits

Finale

We have covered a LOT

- Electricity on wires
- Transistors to gates (AND, OR, ...)
- Combined gates to make circuits
- Connected circuits and registers to build a 1-byte computer

Finale

We have covered a LOT

- Electricity on wires
- Transistors to gates (AND, OR, ...)
- Combined gates to make circuits
- Connected circuits and registers to build a 1-byte computer
- Wrote an ISA for that computer (1-byte instructions, Toy ISA)

Finale

We have covered a LOT

- Electricity on wires
- Transistors to gates (AND, OR, ...)
- Combined gates to make circuits
- Connected circuits and registers to build a 1-byte computer
- Wrote an ISA for that computer (1-byte instructions, Toy ISA)
- Expanded to x86-64 Assembly (saw the binary)

Finale

We have covered a LOT

- Electricity on wires
- Transistors to gates (AND, OR, ...)
- Combined gates to make circuits
- Connected circuits and registers to build a 1-byte computer
- Wrote an ISA for that computer (1-byte instructions, Toy ISA)
- Expanded to x86-64 Assembly (saw the binary)
- Concluded with C (how it compiles and connects with Assembly)

Thanks for a great semester!