Objectives - Wed 1/29/2020

- Code quality
  - Fix an anonymous student's code
- Memory
  - Segments: stack, data segment, text segment
- Memory addresses

© 2020 Alexander J. Quinn. This content is protected and may not be shared, uploaded, or distributed.

1-29-2020

Code quality

Memory

Strings

```
INIT;
while(CONDITION) {

    INCREMENT;
}
```

Use this

```
for(INIT; CONDITION; INCREMENT){
}
```

1-28-2020

Fix indent for whole file:

$$\underbrace{gg}_{\text{to top}} \underbrace{= \boxed{G}}_{\text{auto-indent to bottom of file}}$$

motion

0202-02-1

Memory

Quick storage of bytes

(value ∈ [0, 255])

8 bits

address 0

You may not write here. (segmentation fault)

Can't predict what will be there before you initialize.

Can't predict where your variable will be.

Stack

addr	type*	name*	value	part	fn
200	int	argc	1	args	main(...)
204	char**	argv	→ {"./foo"}		
212	void*	~~where to go after fn returns~~		ret addr	
220	int	n	~~1~~ 5	locals	
224	int*	a_n	220	args	make_5()
232	void*	(somewhere in main)		return address	
		none		locals	

Heap

addr	value	🔒
400		

Data segment

addr	type*	value
600		

- Type and name are not actually stored in memory or executable. Addresses shown are fictional.
- Assume `sizeof(int)==4`
  `sizeof(char)==1`
  `sizeof(void*)==8`
- To show struct types with fields, split the type and name fields. In value field, just write the value of the field. Example →

type	name	value
Point : int,	p . x	5
: int	. y	6