SER 334 A Session

SI Session

Thursday, January 11th 2024

7:00 pm - 8:00 pm MST

Agenda

{

What is SI?

About Me

Session Times & Expectations

Pointers

Course Tips!
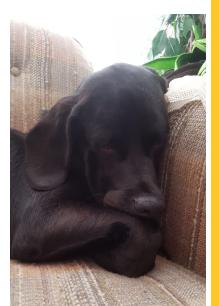
What is SI?

3

About Me
Katie

- Born and raised near Chicago, Illinois
  - Moved around within the Northern Suburbs
  - Currently in Grayslake
- Work at a family-owned Italian restaurant

- Received AS from my local Community College
- Transferred to ASU at the start of 2020
- Majoring in Software Engineering
- Runs in the family
  - Father is a Software Architect
  - Mother is a Quality Assurance Analyst

About Me
Katie

- Hobbies
  - Gaming
  - Football (Go Bears!)
  - Reading
  - Learning
  - Doggos! 😊

SI Session Expectations

Thanks for coming to the **SER 334** SI session. We have a packed agenda and we are going to try to get through as many of our planned example problems as possible. This session will be recorded and shared with others.

- If after this you want to see additional examples, please visit the drop-in tutoring center.
- We will post the link in the chat now and at the end of the session.
- Please keep in mind we are recording this session and it will be made available for you to review 24-48 hours after this session concludes.
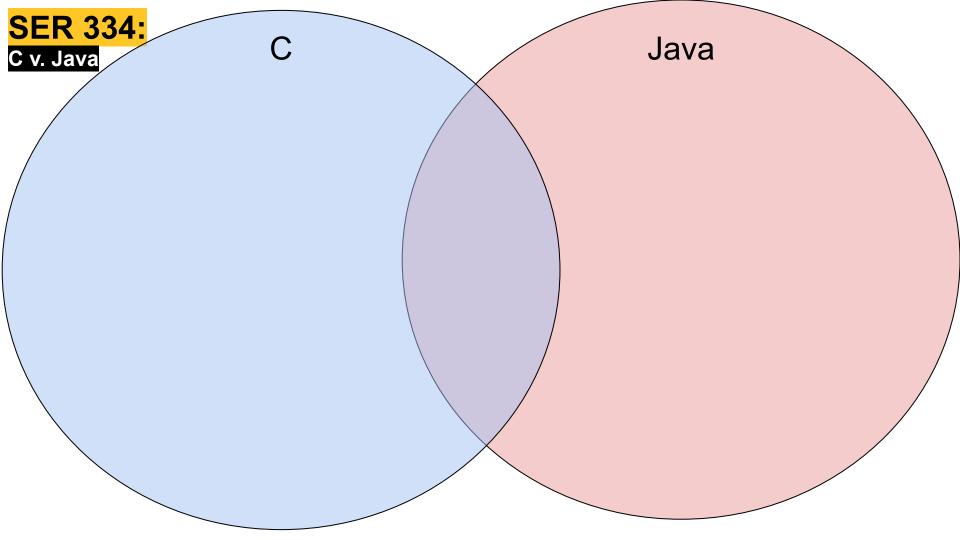- Finally, please be respectful to each other during the session.

Interact with us:
Zoom Features

Zoom Chat

- Use the chat feature to interact with the presenter and respond to presenter's questions.
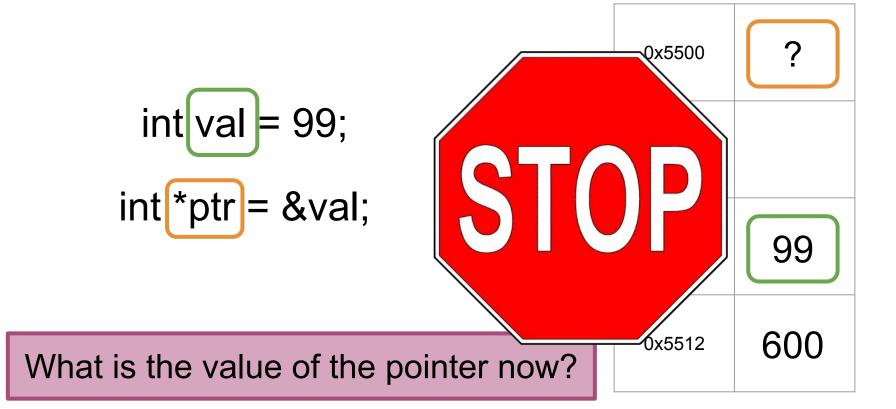- Annotations are encouraged.
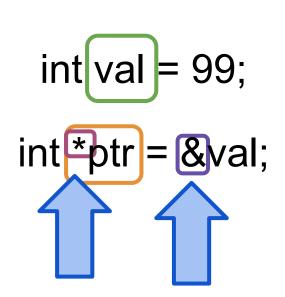
C

Java

What is a pointer?

int *ptr;

Variable

Value is a memory address

What is a pointer?

int val = 99;

int *ptr = &val;

0x5500	?
	99
0x5512	600

What is the value of the pointer now?

Two Pointer Operators

int val = 99;

int *ptr = &val;

0x5500	?
0x5504	
0x5508	99
0x5512	600

Reference or Dereference?
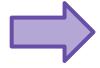
* ➡ Dereference Value at
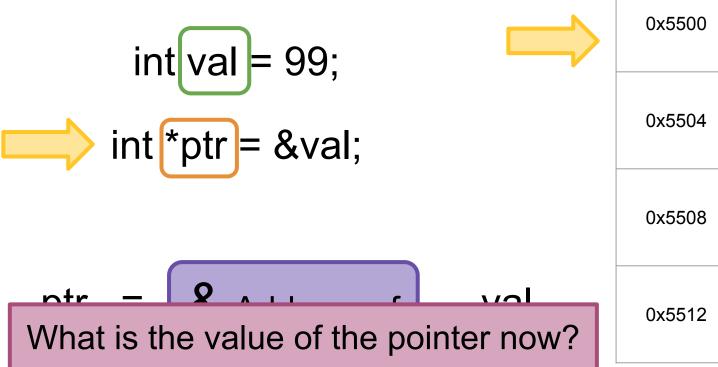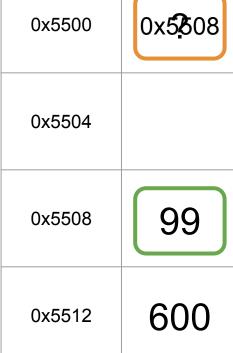
& ➡ Reference Address of

* Value at
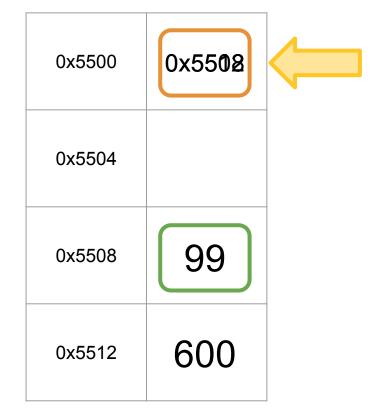
& Address of

int val = 99;

int *ptr = &val;

ptr = **&** Address of val

What is the value of the pointer now?

0x5500	0x5508?
0x5504	
0x5508	99
0x5512	600

* Value at

& Address of

int val = 99;

int *ptr = &val;

➡ ptr = ptr + 4;

0x5500	0x5508 ⬅
0x5504	
0x5508	99
0x5512	600

* Value at

& Address of

int val = 99;

int *ptr = &val;

ptr = ptr + 4;

⇨ *ptr = 5;

The value at

This location

0x5500	0x5512 ⬅
0x5504	
0x5508	99
0x5512	600

* Value at

& Address of

int val = 99;

int *ptr = &val;

ptr = ptr + 4;

➡ *ptr = 5;

The value at

This location

0x5500	0x5512
0x5504	
0x5508	99
0x5512	650
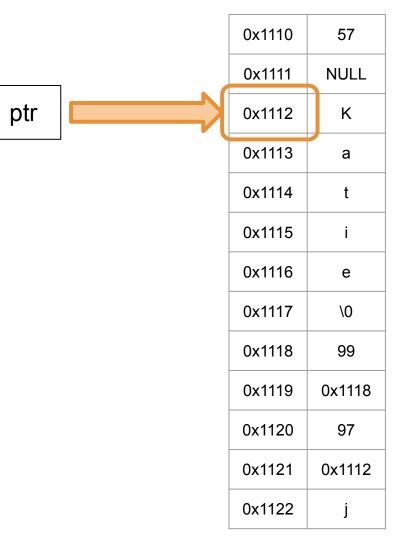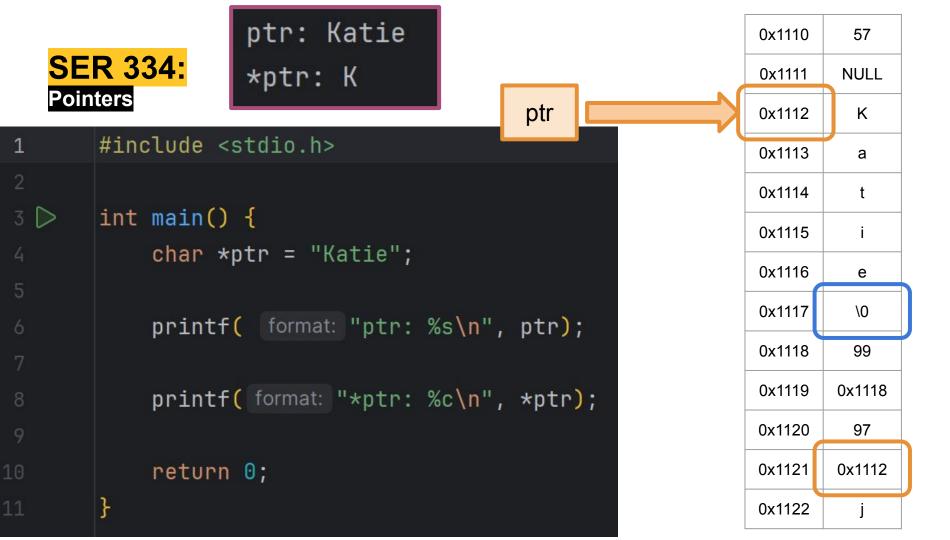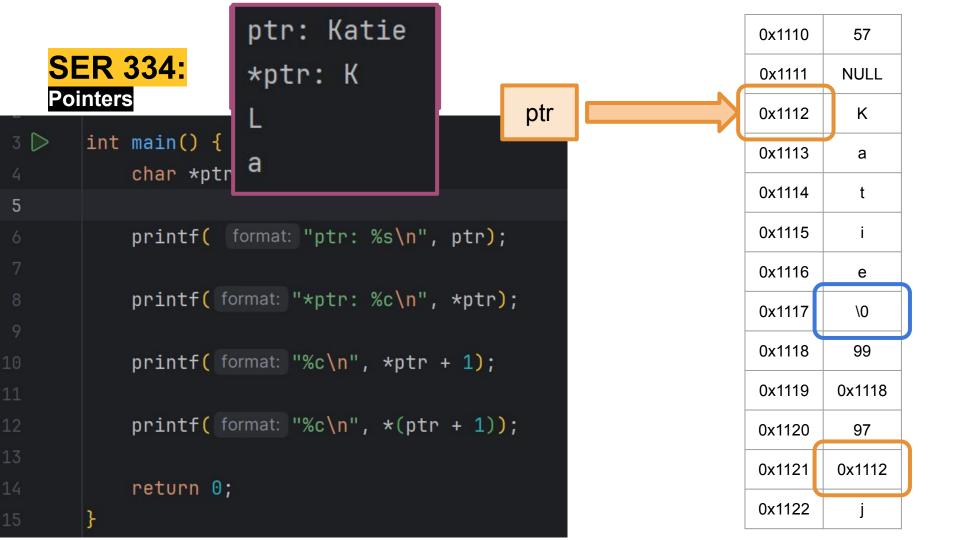
char *ptr = "Katie";

Pointers are variables that store…

ptr

0x1110	57
0x1111	NULL
0x1112	K
0x1113	a
0x1114	t
0x1115	i
0x1116	e
0x1117	\0
0x1118	99
0x1119	0x1118
0x1120	97
0x1121	0x1112
0x1122	j

```
ptr: Katie
*ptr: K
```

0x1110	57
0x1111	NULL
0x1112	K
0x1113	a
0x1114	t
0x1115	i
0x1116	e
0x1117	\0
0x1118	99
0x1119	0x1118
0x1120	97
0x1121	0x1112
0x1122	j

ptr

```c
1    #include <stdio.h>
2
3    int main() {
4        char *ptr = "Katie";
5
6        printf( format: "ptr: %s\n", ptr);
7
8        printf( format: "*ptr: %c\n", *ptr);
9
10       return 0;
11   }
```

```
ptr: Katie
*ptr: K
L
a
```

```
3  ▷   int main() {
4          char *ptr
5
6          printf( format: "ptr: %s\n", ptr);
7
8          printf( format: "*ptr: %c\n", *ptr);
9
10         printf( format: "%c\n", *ptr + 1);
11
12         printf( format: "%c\n", *(ptr + 1));
13
14         return 0;
15     }
```

ptr →

0x1110	57
0x1111	NULL
0x1112	K
0x1113	a
0x1114	t
0x1115	i
0x1116	e
0x1117	\0
0x1118	99
0x1119	0x1118
0x1120	97
0x1121	0x1112
0x1122	j

iPtr →	0x1110	57
	0x1111	NULL
ptr →	0x1112	K
	0x1113	a
	0x1114	t
	0x1115	i
	0x1116	e
	0x1117	\0
	0x1118	99
	0x1119	0x1118
	0x1120	97
	0x1121	0x1112
	0x1122	j

```
iPtr = iPtr + 4;

printf( format: "iPtr: %d\n", iPtr);

printf( format: "*iPtr: %d\n", *iPtr);

*iPtr = 5;

printf( format: "Modified *iPtr: %d\n", *iPtr);
```

SER 334:
Pointers

```c
iPtr = iPtr + 4;

printf( format: "iPtr: %d\n", iPtr);

printf( format: "*iPtr: %d\n", *iPtr);

*iPtr = 5;

printf( format: "Modified *iPtr: %d\n", *iPtr);
```

iPtr →	0x1110	57
	0x1111	NULL
ptr →	0x1112	K
	0x1113	a
	0x1114	t
	0x1115	i
	0x1116	e
	0x1117	\0
	0x1118	99
	0x1119	0x1118
	0x1120	97
	0x1121	0x1112
	0x1122	j

SER 334:
Tips for Success!

- Start assignments ASAP and go one step at a time

- Monitor the slack channels!

- Ask questions right away

- Use the Course Repo - it's there for a reason!

Katie Tip:
    Make a general reference sheet!
    Add notes for concepts you struggle with
    Can then filter the sheet into your note card for the Exam!
You are given a sheet for Unit 1, maybe add onto that!

Upcoming Events

SI Sessions:

- **ASN is Closed Sunday, January 14th and Monday, January 15th**

- Thursday, January 18th at 7:00 pm MST
- Sunday, January 21st at 7:00 pm MST
- Monday, January 22nd at 7:00 pm MST

Review Sessions:

- Exam 1 Review: Thursday, January 25th 7:00 pm - 9:00 pm MST

Questions?

Survey:
http://bit.ly/ASN2324

More Questions?
Check out our other resources!

tutoring.asu.edu

1 -
Go to Zoom

Need help using Zoom?

2 -
View the tutoring schedule

View digital resources

1. **Click on 'Go to Zoom' to log onto our Online Tutoring Center.**
2. **Click on 'View the tutoring schedule' to see when tutors are available for specific courses.**

More Questions?

Check out our other resources!

tutoring.asu.edu/online-study-hub

Don't forget to check out
the Online Study Hub
for additional resources!

Additional Resources

- **[Course Repo](#)**