

SER 321 B Session

SI Session

Sunday, April 1st 2024

7:00 pm - 8:00 pm MST

Agenda



Handling Client Connection

Using Threads

Threading Pitfalls

SI Session Expectations

Thanks for coming to the **SER 321** 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.
 - tutoring.asu.edu
- 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

SER 321

Server Socket

1. Define Params

2. Create Socket

3-5. Mark Socket to Listen

6. Wait for Connection

7. Handle Client Connection

8. Close Client Connection

9. Continue Listening

1

2 & 3-5

9

6

```
public static void main (String args[]) {  
  
    if (args.length != 1) {  
        System.out.println("Expected arguments: <port(int)>");  
        System.exit( status: 1);  
    }  
  
    try {  
        port = Integer.parseInt(args[0]);  
    } catch (NumberFormatException nfe) {  
        System.out.println("[Port|sleepDelay] must be an integer");  
        System.exit( status: 2);  
    }  
  
    try {  
        //open socket  
        ServerSocket serv = new ServerSocket(port);  
        System.out.println("Server ready for connections");  
  
        while (true){  
            System.out.println("Server waiting for a connection");  
            sock = serv.accept(); // blocking wait  
            System.out.println("Client connected");  
            // ...  
        }  
    }  
}
```

SER 321

Server Socket

What needs to be done here?

1. Define Params

2. Create Socket

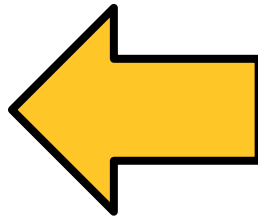
3-5. Mark Socket to Listen

6. Wait for Connection

7. Handle Client Connection

8. Close Client Connection

9. Continue Listening



1

2

3

4

5

SER 321

Server Socket

What needs to be done here?

Is input
from the client
or
to the client ?

1. Define Params

```
// setup the object reading channel  
in = new ObjectInputStream(sock.getInputStream());
```

```
// get output channel  
OutputStream out = sock.getOutputStream();
```

```
// create an object output writer (Java only)  
os = new DataOutputStream(out);
```

1

2

3

4

5

```
clientSock = sock.accept(); // blocking wait  
PrintWriter out = new PrintWriter(clientSock.getOutputStream(), autoFlush: true);  
InputStream input = clientSock.getInputStream();  
System.out.println("Server connected to client");
```

SER 321

Server Socket

What needs to be done here?

```
static void overandout() {  
    try {  
        os.close();  
        in.close();  
        sock.close();  
    } catch (Exception e) {e.printStackTrace();}  
}  
  
try {  
    s = (String) in.readObject();  
} catch (Exception e) {  
    System.out.println("Client disconnect");  
    connected = false;  
    continue;  
}
```

1 Create input/output streams

2

3

4

5

SER 321

Server Socket

What needs to be done here?

```
JSONObject res = isValid(s);

if (res.has(key: "ok")) {
    writeOut(res);
    continue;
}

JSONObject req = new JSONObject(s);

res = testField(req, key: "type");
if (!res.getBoolean(key: "ok")) {
    res = noType(req);
    writeOut(res);
    continue;
}
```

```
public static JSONObject isValid(String json) {
    try {
        static JSONObject testField(JSONObject req, String key){
            JSONObject res = new JSONObject();

            // field does not exist
            if (!req.has(key)){
                res.put("ok", false);
                res.put("message", "Field " + key + " does not exist in request");
                return res;
            }

            return res.put("ok", true);
        }

        return res;
    }
}

return new JSONObject();
}
```

SER 321**Server Socket**

What needs to be done here?

```
int numr = input.read(clientInput, off: 0, buflen);
```

```
String received = new String(clientInput, offset: 0, numr);  
System.out.println("read from client: " + received);  
out.println(received);
```

```
if (req.getString(key: "type").equals("echo")) {  
    res = echo(req);  
} else if (req.getString(key: "type").equals("add")) {  
    res = add(req);  
} else if (req.getString(key: "type").equals("addmany")) {  
    res = addmany(req);  
} else {  
    res = wrongType(req);  
}  
writeOut(res);
```

1 Create input/output streams

2 Check for disconnect

3 Check Protocol

4

5

SER 321

Server Socket

What needs to be done here?

1. Define Params

2. Create Socket

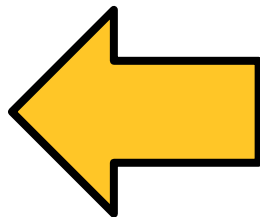
3-5. Mark Socket to Listen

6. Wait for Connection

7. Handle Client Connection

8. Close Client Connection

9. Continue Listening



1 Create input/output streams

2 Check for disconnect

3 Check Protocol

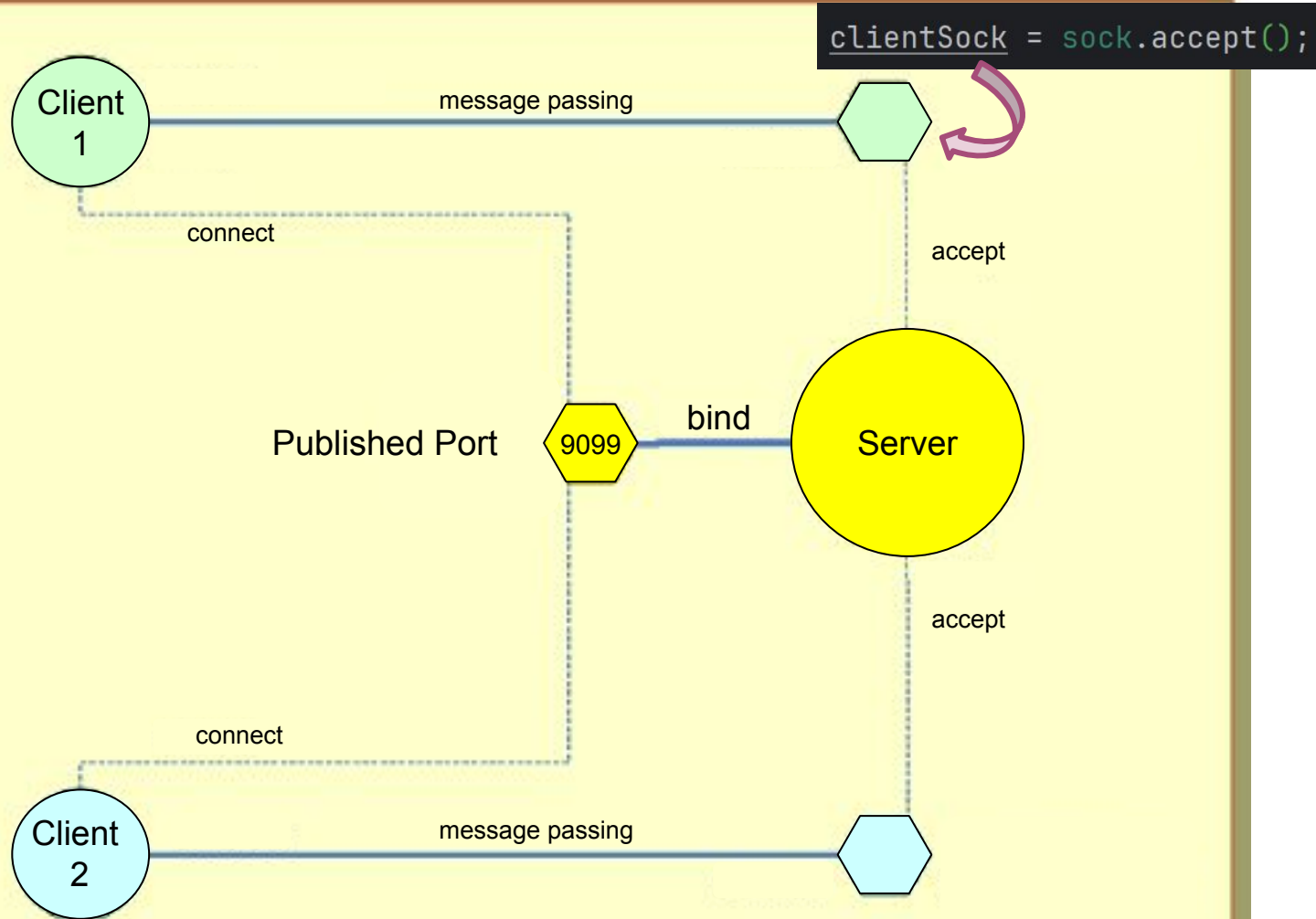
4 Read Headers

5 Handle Accordingly

SER 321

Sockets!

```
> Task :runServer
Server ready for connections
Server is listening on port: 9099
-----
Values of the ServerSocket Object:
Inet Address: 0.0.0.0/0.0.0.0
Local Port: 9099
Server waiting for a connection
Server connected to client
-----
Values of the Client Socket Object
Inet Address: /127.0.0.1
Local Address: /127.0.0.1
Local Port: 9099
Allocated Client Socket (P
<=====--> 75% EXECUTING [31s
> :runServer
```

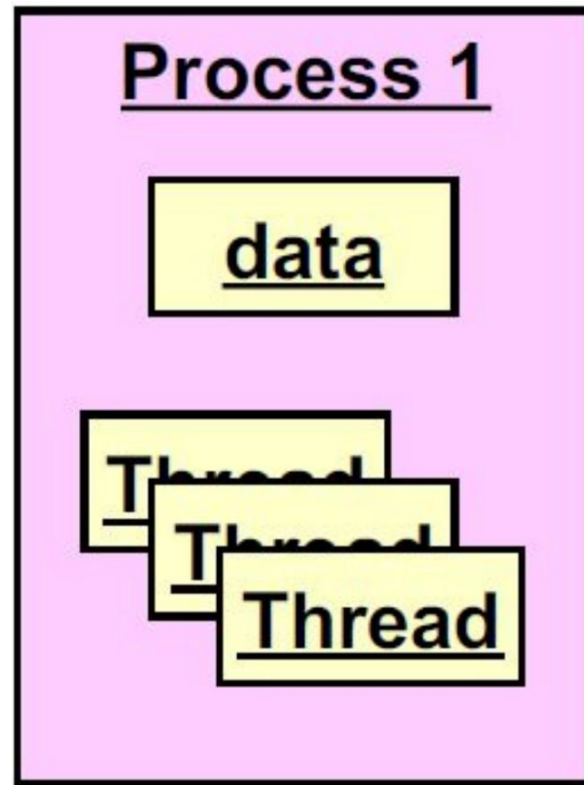


SER 321

Threads

Remember that they exist
within the parent process

What does that imply?



SER 321

Threading Pitfalls

Race Condition

A thread never gains access to the resource it needs

Starvation

A thread is only able to acquire some of the resources it needs

Deadlock

More than one thread accesses a single resource at the same time

SER 321

Threading Pitfalls

Race Condition

A thread never gains access to the resource it needs

Starvation

A thread is only able to acquire some of the resources it needs

Deadlock

More than one thread accesses a single resource at the same time

SER 321
Threading Pitfalls

What does *Spaghetti Consumed* represent?

What does *Thinking* represent?

What does *Hungry* represent?

powered by NetLogo

Dining Philosophers

File: New Revert to Original

Export: NetLogo HTML

Mode: Interactive

Commands and Code: Bottom

model speed

ticks: 6712

num-philosophers

12

setup

go

go once

hungry-chance

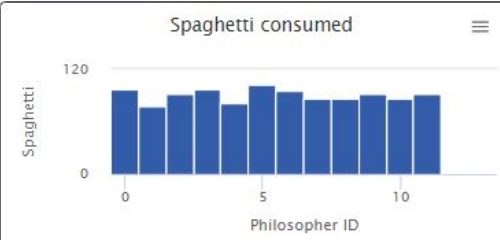
0.5

full-chance


0.5

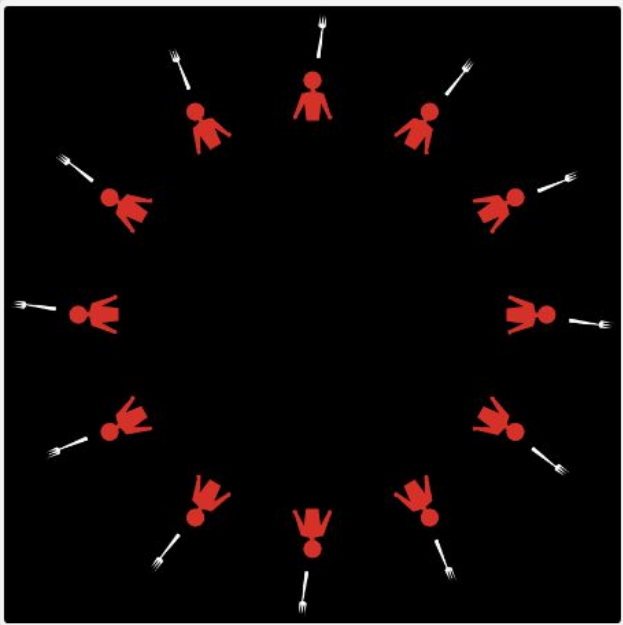
☐ cooperation?

Spaghetti consumed



Resource allocation

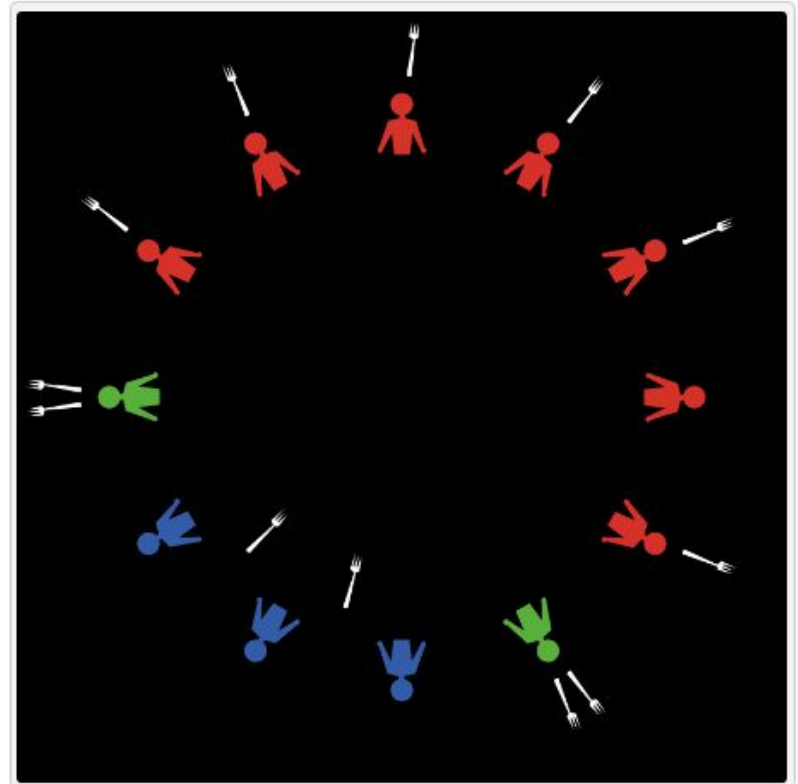




SER 321

Threading Pitfalls

Can we identify any concurrency issues here?



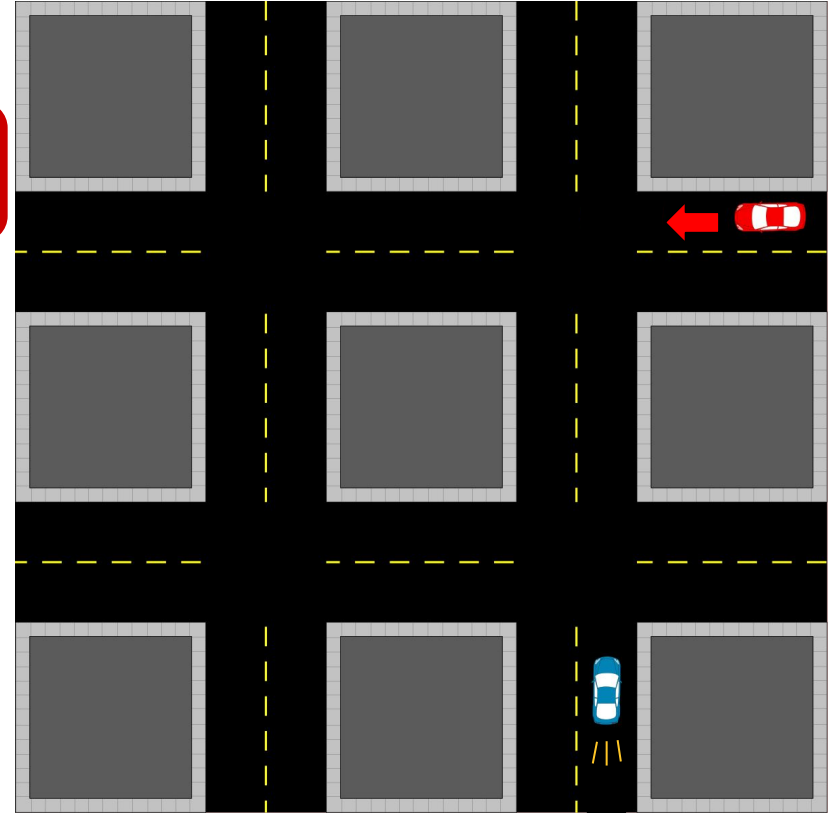
SER 321

Threading Pitfalls

Race Condition

Crash

More than one thread accesses a single resource at once



SER 321

Threading Pitfalls

Race Condition

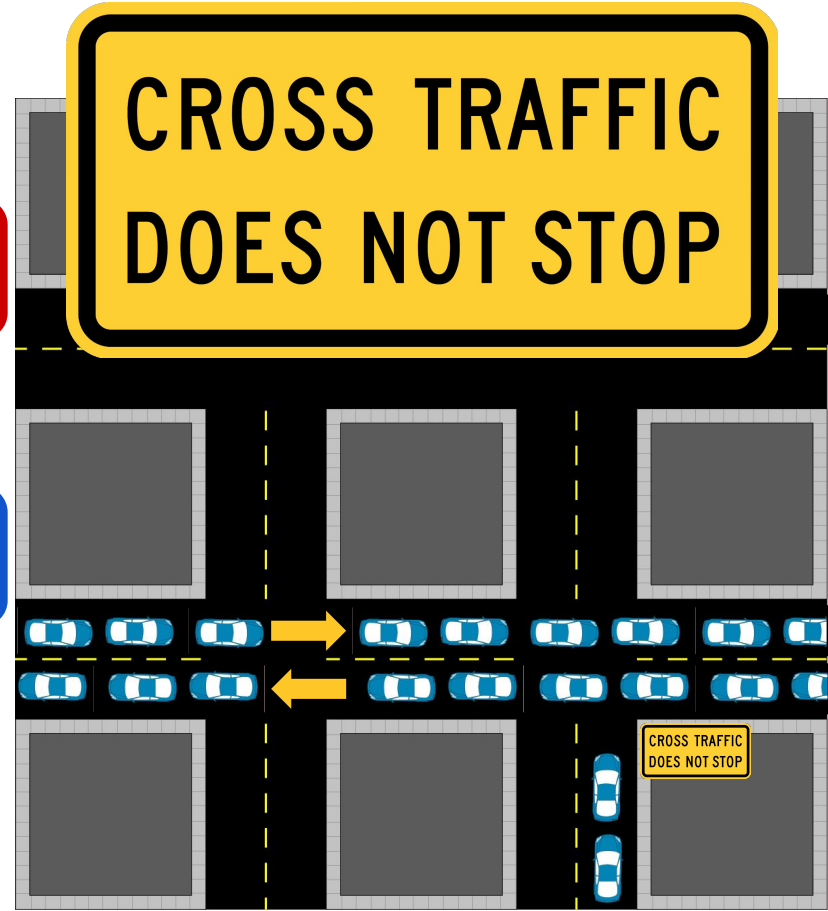
Crash

More than one thread accesses a single resource at once

Starvation

Cross Traffic

A thread never gains access to the resource it needs



SER 321

Threading Pitfalls

Race Condition

Crash

More than one thread accesses a single resource at once

Starvation

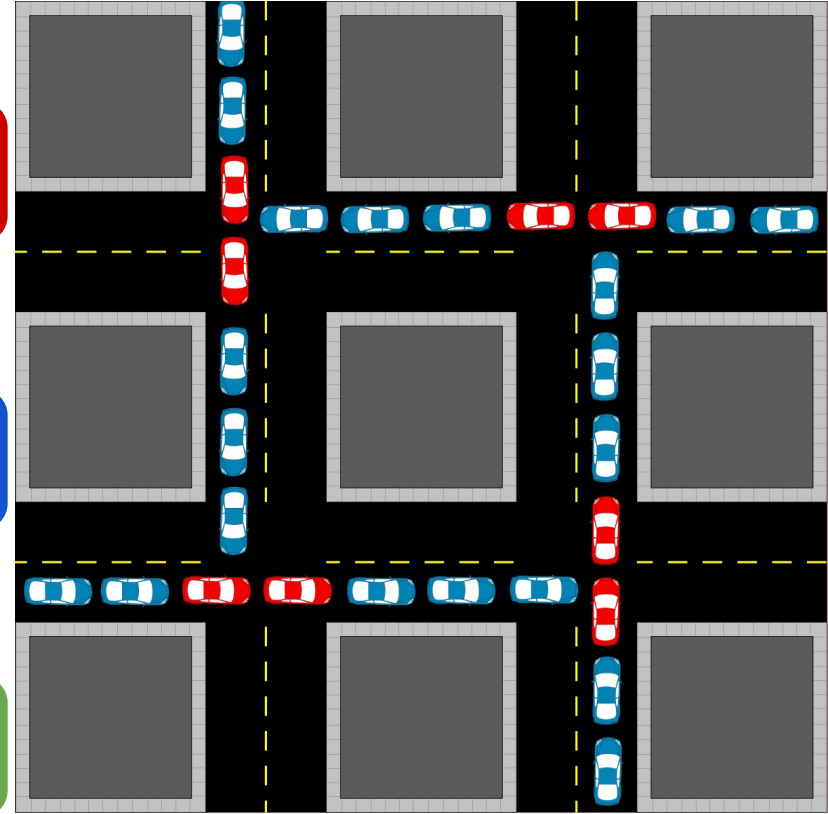
Cross Traffic

A thread never gains access to the resource it needs

Deadlock

Gridlock

A thread is only able to acquire some of the needed resources



SER 321

Scratch Space

Questions?

Survey:

<http://bit.ly/ASN2324>



Upcoming Events

SI Sessions:

- Thursday, April 4th at 7:00 pm MST
- Sunday, April 7th at 7:00 pm MST
- Monday, April 8th at 7:00 pm MST

Review Sessions:

- Sunday, April 21st at 7:00 pm MST
- **Monday, April 25th Session is *cancelled***

More Questions?

Check out our other resources!

tutoring.asu.edu



Academic Support

Academic Support Network (ASN) provides a variety of free services in-person and online to help currently enrolled ASU students succeed academically.

Services



Subject Area Tutoring

Need in-person or online help with math, science, business, or engineering courses? Just hop into our Zoom room or drop into a center for small group tutoring. We'll take it from there.

[Need help using Zoom?](#)

[View the tutoring schedule](#)

[View digital resources](#)

Go to Zoom



Writing Tutoring

Need help with undergraduate or graduate writing assignments? Schedule an in-person or online appointment, access your appointment link, or wait in our drop-in queue.

[Access your appointment link](#)

[Access the drop-in queue](#)

Schedule Appointment



Online Study Hub

Join our online peer communities to connect with your fellow Sun Devils. Engage with our tools to search our bank of resources, videos, and previously asked questions. Or, ask our Tutorbot questions.

Now supporting courses in Math, Science, Business, Engineering, and Writing.

Online Study Hub

1-

Go to Zoom

2-

[Need help using Zoom?](#)

[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

 **Academic Support Network**

 [Services](#)  [Faculty and Staff Resources](#) [About Us](#) 

[University College](#)

Online Study Hub

Online peer communities for students and tutors, YouTube channels, and Tutorbots.



What are online peer communities?

Individual courses have an online peer community that allows you to connect with your peers to post and answer questions and to develop study groups.



How can tutoring center videos help?

Videos can help supplement the learning you're doing in and outside of class and include step-by-step methods for how to understand concepts.



How does the Tutorbot work?

You can ask the Tutorbot questions about course concepts and the Tutorbot will recommend additional resources and examples to help address your questions.

Select a subject

- Any -

Apply



Academic Support Network



[Services](#) 

[Faculty and Staff Resources](#)

[About Us](#) 

[University College](#)

Select a subject

- Any -

Apply

Business

ACC 231

Uses of Accounting Info I

 [Peer Community](#)

ACC 241

Uses of Accounting Info II

 [Peer Community](#)

CIS 105

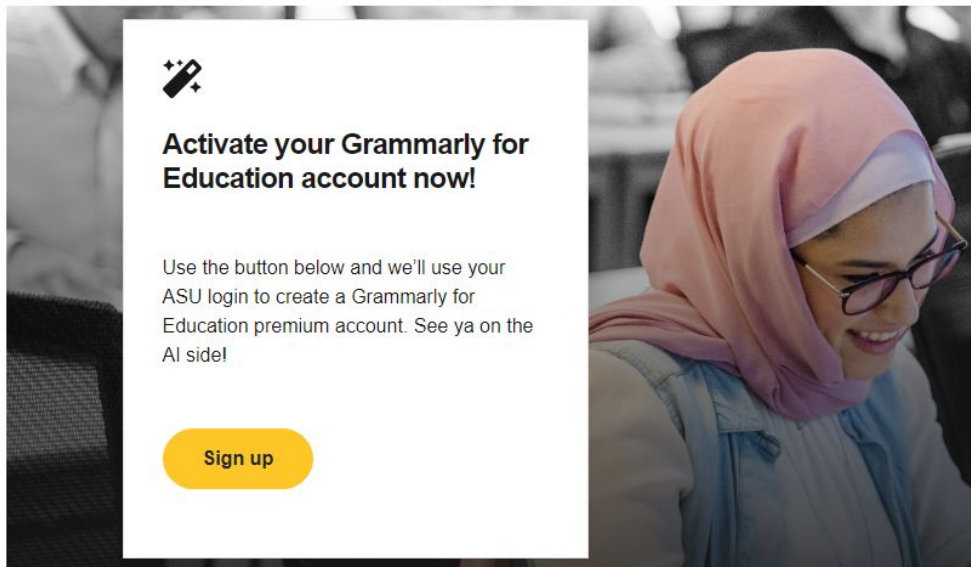
Computer Applications and Information Technology


 [Peer Community](#)

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

Expanded Writing Support Available

Including Grammarly for Education, at no cost!





Activate your Grammarly for Education account now!

Use the button below and we'll use your ASU login to create a Grammarly for Education premium account. See ya on the AI side!

[Sign up](#)



tutoring.asu.edu/expanded-writing-support

*Available slots for this pilot are limited

Additional Resources

- [Course Repo](#)
- [Gradle Documentation](#)
- [GitHub SSH Help](#)
- [Linux Man Pages](#)
- [OSI Interactive](#)
- [MDN HTTP Docs](#)
 - [Requests](#)
 - [Responses](#)
- [JSON Guide](#)
- [org.json Docs](#)
- [javax.swing package API](#)
- [Swing Tutorials](#)
- [Dining Philosophers Interactive](#)
- [Austin G Walters Traffic Comparison](#)