Ice cream parlor

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone. (customers can create clerks out of thin air in this simulation.)
Ice cream parlor

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone. (customers can create clerks out of thin air in this simulation.)

- Make ice cream cone
Ice cream parlor

- **Customer**
  - Order 1-4 ice cream cones
  - Hire a clerk on-demand to make each cone. (Customers can create clerks out of thin air in this simulation.)

- **Clerk**
  - Make ice cream cone
  - Go to manager's office
  - Wake up the manager and present the cone

- **Manager**
  - Go to sleep until a clerk requests an inspection

- **Office of the Manager**
Ice cream parlor

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone. (customers can create clerks out of thin air in this simulation.)

- Make ice cream cone
- Go to manager's office
- Wake up the manager and present the cone
- Wait for manager's approval or rejection

- Go to sleep until a clerk requests an inspection
Ice cream parlor

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone. (Customers can create clerks out of thin air in this simulation.)

- Make ice cream cone
- Go to manager's office
- Wake up the manager and present the cone
- Wait for manager's approval or rejection

- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
- Repeat
Ice cream parlor

Customer:
- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone. (Customers can create clerks out of thin air in this simulation.)

Clerk:
- Make ice cream cone
- Go to manager’s office
- Wake up the manager and present the cone
- Wait for manager’s approval or rejection
- Repeat if rejected

Manager:
- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
- Repeat
Ice cream parlor

**Customer**
- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone. (Customers can create clerks out of thin air in this simulation.)

**Clerk**
- Make ice cream cone
- Go to manager's office
- Wake up the manager and present the cone
- Wait for manager's approval or rejection
- Repeat if rejected

**Manager**
- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
- Repeat
Ice cream parlor

**customer**
- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone.
- Wait for all clerks to finish
- Get in line to see cashier
- Wait until cashier says we’re done

**clerk**
- Make ice cream cone
- Go to manager’s office
- Wake up the manager and present the cone
- Wait for manager’s approval or rejection
- Repeat if rejected

**manager**
- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
- Repeat

**cashier**
- Wait for customer to get in line
- Ring up that customer
- Tell the customer they can leave

**electrician**
- Wait until all customers have left
- Fix electrical wiring
Implementation

- For this example, no need to actually pass ice cream cones around
- Going to focus on the synchronization between threads
- Goals:
  - What synchronization primitive should we choose in each situation?
  - How can we break this problem down and implement it in small parts?
Customer <-> Clerk

- **Customer:**
  - How to summon clerks on demand?
  - How to wait until clerks are finished?
- **Can we implement/test this without implementing the clerk yet?**
- **Clerk:**
  - Call `makeCone(coneId, customerId);`

- **Order 1-4 ice cream cones**
- **Hire a clerk on-demand to make each cone**
- **Wait for all clerks to finish making the cones**
- **Make ice cream cone**
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Go to sleep until a clerk requests an inspection
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Go to sleep until a clerk requests an inspection
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Wake up the manager and present the cone

- Go to sleep until a clerk requests an inspection
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Wake up the manager and present the cone
- Go to sleep until a clerk requests an inspection
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Wake up the manager and present the cone
- Wait for manager's approval or rejection

- Go to sleep until a clerk requests an inspection
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Wake up the manager and present the cone
- Wait for manager's approval or rejection
- Go to sleep until a clerk requests an inspection
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Wake up the manager and present the cone
- Wait for manager's approval or rejection

- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Wake up the manager and present the cone
- Wait for manager's approval or rejection

- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
Clerk <-> Manager

- Make ice cream cone
- Go to manager's office. **Only one clerk in the office at a time.**
- Wake up the manager and present the cone
- Wait for manager's approval or rejection

- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
Clerk <-> Manager

Questions:
- How to have only one clerk in the office at a time?
- How to have the manager sleep until a clerk needs an inspection?
- How to have the clerk wait until the inspection is complete?
- How to communicate the results of the inspection (passed/failed)?

How can we implement/test as little as possible at a time?

- Make ice cream cone
- Go to manager's office. Only one clerk in the office at a time.
- Wake up the manager and present the cone
- Wait for manager's approval or rejection
- If rejected, remake the cone and repeat

- Go to sleep until a clerk requests an inspection
- Inspect the cone
- Inform the clerk of the decision
- Repeat
Customer <-> Cashier

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone
- Get in line to see cashier

- Wait for customer to get in line
Customer <-> Cashier

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone
- Get in line to see cashier
- Wait until cashier says we’re done

- Wait for customer to get in line
Customer <-> Cashier

• Order 1-4 ice cream cones
• Hire a clerk on-demand to make each cone
• Get in line to see cashier
• Wait until cashier says we’re done

• Wait for customer to get in line
• Ring up that customer
• Tell the customer they can leave
Customer <-> Cashier

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone
- Get in line to see cashier
- Wait until cashier says we’re done

- Wait for customer to get in line
- Ring up that customer
- Tell the customer they can leave
Questions:
- How should customers join the line / store their ordering in line? (How does the cashier know who to help first?)
- How should the cashier wait until a customer has joined the line?
- How should the cashier notify a specific customer that they are finished?
Electrician <-> Customers

- Order 1-4 ice cream cones
- Hire a clerk on-demand to make each cone
- Get in line to see cashier
- Wait until cashier says we’re done

- Wait until all customers have left
- Fix electrical wiring

Questions:
- How should electricians wait until all customers have left the store?