Outline

1. Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2. The `idle()` Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
1. Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2. The `idle()` Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
Animation involves moving objects around the scene.

As in any movie, we simulate continuous movement by rapidly drawing a series of frames in which the objects are drawn in slightly different positions.

The parameters that we control, for each object, are:
- The starting and ending positions.
- The duration of the animation.

We assume that the transition is linear.
Implementing Animation

```c
float start_time = clock(); // Current time
float elapsed_time = (clock() - start_time)/CLOCKS_PER_SEC;
float duration = 4.0f;       // Duration of animation
float curr_pos = start_pos
                + elapsed_time/duration*(end_pos - start_pos);
```
Outline

1. Implementing Animation
   - The Water Level
     - The Paddles
     - The Gates

2. The `idle()` Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
float duration = 3.0f; // Duration of animation
float water_level = WATER_LOW + elapsed_time/duration*(WATER_HIGH - WATER_LOW);

- Raising the water level is very simple.
switch (fkey)
{
    :;
    case 12: // Raise the water level
        if (water_level == WATER_HIGH)
            elapsed_time = duration;
        else
            water_level = WATER_LOW
                    + elapsed_time/duration*(WATER_HIGH - WATER_LOW);
        break;
    :;
}"
Outline

1. Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2. The `idle()` Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
float duration = 1.0f;
float paddle_angle = PADDLE_CLOSED_ANGLE
                 + elapsed_time/duration
                 *(PADDLE_OPEN_ANGLE - PADDLE_CLOSED_ANGLE);

- Opening a paddle is also very simple.
- The variable is the angle.
To rotate the paddles, we must include a `rotate()` instruction in the draw function.

How do we rotate a paddle about its center?
1. Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2. The idle() Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
Rotating a gate is similar to rotating a paddle.

Except that \texttt{PADDLE\_CLOSED\_ANGLE} is not 0°.
Outline

1. Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2. The `idle()` Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
The \texttt{idle()} Function

```c
void idle()
{
    float elapsed_time = (clock() - start_time)/CLOCKS_PER_SEC;
    if (animate)
    {
        // Update positions
        :
        if (elapsed_time >= duration)
            animate = \texttt{false};
    }
}
```

- Use a boolean variable \texttt{animate} to tell whether or not animation is in progress.
Outline

1. Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2. The `idle()` Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
Opening paddles may or may not be followed by the water level rising or falling, depending on which paddles were opened and current water level.
Animating Sequences

- Opening paddles may or may not be followed by the water level rising or falling, depending on which paddles were opened and current water level.
- If we open the upper paddles and the water is already high, then we should skip animating the water rising.
Opening paddles may or may not be followed by the water level rising or falling, depending on which paddles were opened and current water level.

If we open the upper paddles and the water is already high, then we should skip animating the water rising.

If we open the lower paddles and the water is already low, then we should skip animating the water falling.
Opening paddles may or may not be followed by the water level rising or falling, depending on which paddles were opened and current water level.

If we open the upper paddles and the water is already high, then we should skip animating the water rising.

If we open the lower paddles and the water is already low, then we should skip animating the water falling.

Indeed, the paddles themselves may already be open, in which case we do not animate their opening either.
Opening paddles may or may not be followed by the water level rising or falling, depending on which paddles were opened and current water level.

If we open the upper paddles and the water is already high, then we should skip animating the water rising.

If we open the lower paddles and the water is already low, then we should skip animating the water falling.

Indeed, the paddles themselves may already be open, in which case we do not animate their opening either.

Furthermore, the water level should not change until after the paddles have been opened.
```
float paddle_dur = 1.0f; // Open paddles in 1 sec
float water_dur = 3.0f; // Raise water in 3 sec

case 0: // Open the upper paddles
    // Check whether paddles are open and water is high
    if (upper_paddle_angle == PADDLE_OPEN_ANGLE
        && water_level == WATER_HIGH)
        elapsed_time = duration;
    // Check whether paddle animation is in progress
    else if (upper_paddle_angle < PADDLE_OPEN_ANGLE
        && elapsed_time <= paddle_dur)
        paddle_angle = PADDLE_CLOSED_ANGLE + ...;
    // Check whether it's time to animate the water level
    else if (elapsed_time > paddle_dur)
        water_level = WATER_LOW + ...;
```

How do we calculate the current water level?
Outline

1. Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2. The idle() Function

3. Animating Sequences

4. Moving the Boat

5. Assignment 19

6. Assignment
Moving the Boat

- Let F9 move the boat downstream one “unit” and F10 move the boat upstream one “unit.”
- A unit is a distance that will move the boat out of the lock (either way) if it is in the lock.
- Index the possible boat positions, with
  - \( \text{pos} \equiv 0 \) signifying in the lock.
  - \( \text{pos} > 0 \) signifying downstream.
  - \( \text{pos} < 0 \) signifying upstream.
There are constraints on the boat’s movements.

Downstream movement (F9)
- If the boat is upstream ($pos < 0$), then . . .
- If the boat is in the lock ($pos == 0$), then . . .
- If the boat is downstream ($pos > 0$), then . . .

The upstream constraints are similar.
1 Implementing Animation
   - The Water Level
   - The Paddles
   - The Gates

2 The idle() Function

3 Animating Sequences

4 Moving the Boat

5 Assignment 19

6 Assignment
Find a stone or masonry texture.
Convert it to the .dds format.
Paste the texture onto the lock wall, including the drain and the wall ends.
Paste a separate texture onto the paddle.