Non-linear Motion

Elements of Graphics
CS324e
What is Continuity?

- Defines level of smoothness along a curve
- Types of continuity:
  - C0: curves are joined
  - C1: first derivatives of curves are continuous
  - C2: second derivatives of curves are continuous

(Wikipedia)
Higher Continuity

- Higher-order continuities have advantages in both geometric modeling and in animation
- Some overhead, since additional points of data are required
- Splines are piecewise polynomials (multiple, connected lower-degree polynomial functions) with high continuity
Quadratic Bezier Curve
Question

- How does the quadratic Bezier curve interpolate between points?
Cubic Bezier Curve
Question

❖ How does the cubic Bezier curve interpolate between points?
Combining Bezier Curves

Bezier curves can be joined to form a B-spline!

Sequence of Bezier curves and control points that form a B-spline (Wikipedia)
Other Interpolation Functions

- Interpolation calculates intermediate values between targets.
- Interpolation along arbitrary functions generates a range of behaviors.
Sine Waves

- Sine (and cosine) equations model a periodic relationship between radians and sine/cosine value.
- Sine waves have an angle ($\Theta$), amplitude ($a$) and phase ($p$).

<table>
<thead>
<tr>
<th>Degrees</th>
<th>0</th>
<th>90</th>
<th>180</th>
<th>270</th>
<th>360</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radians</td>
<td>0</td>
<td>$\pi/2$</td>
<td>$\pi$</td>
<td>$\pi+\pi/2$</td>
<td>$2\pi$</td>
</tr>
<tr>
<td>Decimal radians</td>
<td>0</td>
<td>1.57</td>
<td>3.14</td>
<td>4.71</td>
<td>6.28</td>
</tr>
<tr>
<td>Constants</td>
<td>0</td>
<td>HALF_PI</td>
<td>PI</td>
<td>PI+HALF_PI</td>
<td>TWO_PI</td>
</tr>
</tbody>
</table>

![Sine wave graph](image_url)
Using Sine

- \( \sin(\Theta) \) outputs a value between -1 and 1 based on \( \Theta \) (between 0 and 2PI)
- Amplitude \( a \) magnifies the value of the sine peak
- Phase \( p \) controls where the oscillation cycle begins
- Sine can control rate of motion (frequency) as well as direction of motion
If \( \sin(\Theta) \) and \( \cos(\Theta) \) are applied to the \( x \) and \( y \) position of an object respectively, what will that object’s motion be?
Easing

- Easing allows movement between two values at nonlinear increments
  - Objects can accelerate/decelerate as they approach the target
- Equation determines the fraction of the distance between the object’s current and target positions that the object moves
Ease-out Example

```java
float x = 0.0;
float easing = 0.05;
float targetX = 400;
void setup() {size(500, 500);}
void draw() {
    x += (targetX - x) * easing;
    ellipse(x, 250, 50, 50);
}
```
We can also think of ease-out as a \textit{lerp} operation over non-even time increments.
Types of Ease Functions

- Robert Penner defined a range of easing equations: http://gizma.com/easing/
- These equations are based on change in value over time
  - Ease in
  - Ease out
  - Ease in/out
- Basic functions are linear, quadratic, cubic, quartic and quintic equations
The graphics featured here represent the transitions that can be used on calls to Tweener's `easing()` and `addCall()` methods to create different easing effects on animations. They are based on Robert Penner's original easing equations. The `linear` transition (seen to the left) is what you would expect of a normal tweening (with no easing at all). The rest of the options have varying easing curves. The default on Tweener is `easeOutExpo`.

(https://code.google.com/archive/p/tweener/)
Hands-on: Linear and Non-Linear Motion

❖ Today’s activities:

1. Recreate Processing’s `lerp()` method that works for values for \( t \) between 0 and 1

2. Rewrite the ease-out function using the `lerp` method

3. Use \( \sin(\Theta) \) to oscillate a ball back and forth from a point then use \( \sin(\Theta) \) to circle a ball around a point

4. Bonus: Create an ease-in function using the `lerp` method. Note that you will need to track the total distance between start and end position as well as the current distance to the end position to do this