Computer Graphics

CS361
Who am I?

- Dr. Barry Wittman
- Education:
  - PhD and MS in Computer Science, Purdue University
  - BS in Computer Science, Morehouse College
- Hobbies:
  - Reading, writing
  - Enjoying ethnic cuisine
  - DJing
  - Lockpicking
How can you reach me?

- **E-mail:** wittmanb@etown.edu
- **Office:** Esbenshade 284B
- **Phone:** (717) 361-4761
- **Office hours:**
  - MWF 11:00am – 12:00pm
  - MWF 2:00 – 3:20pm
  - T 1:00 – 3:00pm
  - And by appointment
- **Website:**
  
  http://users.etown.edu/w/wittmanb/
Why are we here?

- What’s the purpose of this class?
- What do you want to get out of it?
- Do you want to be here?
Course Overview
Textbook

- Tomas Akenine-Möller, Eric Haines, and Naty Hoffman
- *Real-Time Rendering*
  - Third edition, 2008, AK Peters
  - ISBN-10: 1568814240
- Amazing book: The 3D Bible
A note about the book...

- It's really awesome
- It could be used for a graduate course in graphics
- Virtually everyone working in video game programming has a copy of this book
- It is API agnostic
  - Although it mentions OpenGL and DirectX, it is not focused on helping you use either
You have to read the book

- You are expected to read the material before class
- If you're not prepared, you will be asked to leave
  - You will not be able to lecture about the subject
  - You will forfeit the opportunity to take quizzes
  - Much more importantly, you will forfeit the education you have paid around **$100 per class meeting** to get
Computer graphics is a huge area
This course is going to be focused on the real time graphics used in video games
Many of the same ideas can be applied to the graphics in Hollywood movies and scientific visualization
We're going to use MonoGame which is a C# wrapper for the DirectX library

Unfortunately, I'm going to be learning it with you, since I had to change the API since last time
Topics to be covered

- Graphics Basics
- Rendering Pipeline
- GPU Architecture
- Review of Linear Algebra
- Transforms
- Visual Appearance
- Texturing
- Advanced Shading
- Environmental Lighting
- Global Illumination
- Image Based Effects
- Polygonal Techniques
- Intersection Test Methods
- Collision Detection
More information

- For more information, visit the webpage:
  http://users.etown.edu/w/wittmanb/cs361

- The webpage will contain:
  - The most current schedule
  - Notes available for download
  - Reminders about projects and exams
  - Syllabus (you can request a printed copy if you like)
  - Detailed policies and guidelines
Four projects

- 45% of your grade will be four equally weighted projects.
- Each will focus on a different area from the course:
  - Color and 2D manipulations
  - 3D model renderer
  - Interactive 3D game
  - Game with collision detection
- You will work on each project in two-person teams.
Teams

- All projects are done in teams of two
- You may pick your partners
  - But you have to have a different partner for each project!
- One member of the team is the leader
  - The leader must tell me who is on the team before the project is assigned
  - I will copy assignments from the leader’s class Canvas account
Turning in projects

- Projects must be turned in by zipping them up and uploading them to Canvas before the deadline.
- Do not put projects in your public directories.
- Late projects will not be accepted.
  - Exception: Each team will have 3 grace days.
  - You can use the 3 grace days together or separately as extensions for your projects.
  - You must inform me before the deadline that you are going to use grace days.
  - If two people in a team don't have the same number of grace days, the number of days they will have available will be the maximum of those remaining for either teammate.
- Projects that don't compile get 0 points.
Assignments
Assignments

- 10% of your grade will be from homework assignments, given roughly once every three weeks
- About half of the assignments will be written, and about half will be coding
- There will be 5 assignments total
- Assignments will not be due on weeks when projects are due
- Each written assignment typeset in LaTeX is worth 1% extra credit applied to your final grade
Impromptu Student Lectures
5% of your grade will be lectures that you give.
You will give two to three of these lectures at any time during the semester, without any warning.
You will have 3 to 5 minutes in which you must present the material to be read for that day.
- Students are encouraged to ask questions.
There is no better way to learn material than by teaching it.
Polishing public speaking skills is never a bad thing.
Quizzes
Pop Quizzes

- 5% of your grade will be pop quizzes
- These quizzes will be based on material covered in the previous one or two lectures
- They will be graded leniently
- They are useful for these reasons:
  1. Informing me of your understanding
  2. Feedback to you about your understanding
  3. Easy points for you
  4. Attendance
Course Schedule
<table>
<thead>
<tr>
<th>Week</th>
<th>Starting</th>
<th>Topics</th>
<th>Chapters</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01/16/17</td>
<td>Graphics basics</td>
<td>Notes</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>01/23/17</td>
<td>Rendering pipeline</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>01/30/17</td>
<td>GPU architecture</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>02/06/17</td>
<td>Linear algebra review</td>
<td>Appendices A and B</td>
<td>Project 1 Due</td>
</tr>
<tr>
<td>5</td>
<td>02/13/17</td>
<td>Transforms</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>02/20/17</td>
<td>Visual appearance</td>
<td>5</td>
<td>Exam 1</td>
</tr>
<tr>
<td>7</td>
<td>02/27/17</td>
<td>Texturing</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>03/06/17</td>
<td>Spring Break</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>03/13/17</td>
<td>Advanced shading</td>
<td>7</td>
<td>Project 2 Due</td>
</tr>
<tr>
<td>9</td>
<td>03/20/17</td>
<td>Environmental lighting</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>03/27/17</td>
<td>Global illumination</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>04/03/17</td>
<td>Image-based effects</td>
<td>10</td>
<td>Project 3 Due</td>
</tr>
<tr>
<td>12</td>
<td>04/10/17</td>
<td>Polygonal techniques</td>
<td>12</td>
<td>Exam 2</td>
</tr>
<tr>
<td>13</td>
<td>04/17/17</td>
<td>Intersection test methods</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>04/24/17</td>
<td>Collision detection</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>05/01/17</td>
<td>Review</td>
<td>All material</td>
<td>Project 4 Due</td>
</tr>
</tbody>
</table>
Exams

- There will be two equally weighted in-class exams totaling 20% of your final grade
  - Exam 1: 02/24/2017
  - Exam 2: 04/10/2017
- The final exam will be worth another 15% of your grade
  - Final: 11:00am – 2:00pm 05/08/2017
Project schedule

- Project 1:    Tentatively due 02/10/2017
- Project 2:    Tentatively due 03/17/2017
- Project 3:    Tentatively due 04/07/2017
- Project 4:    Tentatively due 05/05/2017
Policies
Grading breakdown

- 45% • Four projects
- 10% • Assignments
- 5% • Quizzes
- 5% • Impromptu student lectures
- 20% • Two equally weighted midterm exams
- 15% • Final exam
# Grading scale

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B-</th>
<th>80-82</th>
<th>D+</th>
<th>67-69</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>93-100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A-</td>
<td>90-92</td>
<td>C+</td>
<td>77-79</td>
<td>D</td>
<td>63-66</td>
</tr>
<tr>
<td>B+</td>
<td>87-89</td>
<td>C</td>
<td>73-76</td>
<td>D-</td>
<td>60-62</td>
</tr>
<tr>
<td>B</td>
<td>83-86</td>
<td>C-</td>
<td>70-72</td>
<td>F</td>
<td>0-59</td>
</tr>
</tbody>
</table>
You are expected to attend class
You are expected to have read the material we are going to cover before class
Missed quizzes cannot be made up
If you miss a day without an excuse, you might get a zero for a lecture you should have given that day
Exams must be made up before the scheduled time, for excused absences
R-E-S-P-E-C-T

- I hate having a slide like this
- I ask for respect for your classmates and for me
- You are smart enough to figure out what that means
- A few specific points:
  - Silence communication devices
  - Don’t use the computers except when explicitly asked to
  - No food or drink in the lab
Computer usage

- Research shows that it is nearly impossible to do two things at the same time (e.g. use Facebook and listen to a lecture)
- For your own good, I will enforce this by taking 1% of your final grade every time I catch you using your computer for anything other than course exercises
Academic dishonesty

- Don’t cheat
- **First offense:**
  - I will give you a zero for the assignment, then lower your final letter grade for the course by one full grade
- **Second offense:**
  - I will fail you for the course and try to kick you out of Elizabethtown College
- Refer to the Student Handbook for the official policy
- Ask me if you have questions or concerns
- **You are never allowed to look at another team's code**
- **I will use tools that automatically test code for similarity**
Elizabethtown College welcomes otherwise qualified students with disabilities to participate in all of its courses, programs, services, and activities. If you have a documented disability and would like to request accommodations in order to access course material, activities, or requirements, please contact the Director of Disability Services, Lynne Davies, by phone (361-1227) or e-mail daviesl@etown.edu. If your documentation meets the college’s documentation guidelines, you will be given a letter from Disability Services for each of your professors. Students experiencing certain documented temporary conditions, such as post-concussive symptoms, may also qualify for temporary academic accommodations and adjustments. As early as possible in the semester, set up an appointment to meet with me, the instructor, to discuss the academic adjustments specified in your accommodations letter as they pertain to my class.
Color
Visible light is a form of electromagnetic radiation. We could think of it as a wave with a specific frequency. That is a useful way to think of sound. It seems sort of cumbersome for light. Color theorists have discovered that we can represent most visible colors as a combination of a small number of set colors.
One system for representing color is **RGB**. With **Red**, **Green**, and **Blue** components, you can combine them to make most visible colors. Combining colors is an additive process:

- With no colors, the background is black
- Adding colors never makes a darker color
- Pure **Red** added to pure **Green** added to pure **Blue** makes **White**

**RGB** is a good model for computer screens.
Everything is a lie

- RGB allows many colors to be represented, but not all.
- The gray area in this picture shows the visible spectrum.
- However, only the colors inside the triangle can be represented using RGB.
CMYK stands for **Cyan, Magenta, Yellow**, and **Key** (Black)

CMYK is another color system, but it’s a **subtractive** system

- With no colors, the background is white
- Adding colors never makes a lighter color
- Pure Cyan added to pure Magenta added to pure Yellow makes **Black**

CMYK is useful for printing, not for computer screens
## Example colors (using a 0-255 system)

<table>
<thead>
<tr>
<th>Color</th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Red</td>
<td>255</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Green</td>
<td>0</td>
<td>255</td>
<td>0</td>
</tr>
<tr>
<td>Blue</td>
<td>0</td>
<td>0</td>
<td>255</td>
</tr>
<tr>
<td>Orange</td>
<td>255</td>
<td>165</td>
<td>0</td>
</tr>
<tr>
<td>Gray</td>
<td>128</td>
<td>128</td>
<td>128</td>
</tr>
<tr>
<td>Cyan</td>
<td>0</td>
<td>255</td>
<td>255</td>
</tr>
<tr>
<td>Magenta</td>
<td>255</td>
<td>0</td>
<td>255</td>
</tr>
<tr>
<td>Yellow</td>
<td>255</td>
<td>255</td>
<td>0</td>
</tr>
<tr>
<td>White</td>
<td>255</td>
<td>255</td>
<td>255</td>
</tr>
</tbody>
</table>
If the R, G, B values happen to be the same, the color is a shade of gray
- 255, 255, 255 = White
- 128, 128, 128 = Gray
- 0, 0, 0 = Black

To convert a color to a shade of gray, use the following formula:
- Value = \(0.3R + 0.59G + 0.11B\)

Based on the way the human eye perceives colors as light intensities
Brightness and Contrast

- We can adjust the brightness of a picture by multiplying each pixel's R, G, and B value by a scalar $b$
  - $b \in [0, 1)$ darkens
  - $b \in (1, \infty)$ brightens
- We can adjust the contrast of a picture by multiplying each pixel's R, G, and B value by a scalar $c$ and then adding $-128c + 128$ to the value
  - $c \in [0, 1)$ decreases contrast
  - $c \in (1, \infty)$ increases contrast
- After adjustments, values must be clamped to the range $[0, 255]$ (or whatever the range is)
Other color models

- RGB is a reasonable model for representing color
- However, it's hard for humans to use RGB to pick the color they want
- HSL and HSV are two models often used for color pickers inside of applications like Photoshop
- HSL
  - Hue (which color)
  - Saturation (how colorful the color is)
  - Lightness (how dark the color is)
- Hue is represented as an angle between 0° and 360°
- Saturation and lightness are often given between 0 and 1
**HSV**

- HSV
  - Hue (which color)
  - Saturation (how colorful the color is)
  - Value (how bright the color is)
- Hue is represented as an angle between $0^\circ$ and $360^\circ$
- Saturation and value are often given between 0 and 1
- Saturation in HSV is not the same as in HSL
Converting from RGB to HSV

Assume the RGB values are in \([0,1]\)

\[
M = \max(R, G, B)
\]

\[
m = \min(R, G, B)
\]

\[
C = M - m
\]

\[
H' = \begin{cases} 
\text{undefined} & C = 0 \\
\frac{G - B}{C} \mod 6 & M = R \\
\frac{B - R}{C} + 2 & M = G \\
\frac{R - G}{C} + 4 & M = B
\end{cases}
\]

\[
H = 60^\circ \times H'
\]

\[
V = M
\]

\[
S = \begin{cases} 
0 & C = 0 \\
\frac{C}{V} & \text{otherwise}
\end{cases}
\]
Converting from HSV to RGB

\[ C = V \times S \]

\[ H' = \frac{H}{60^\circ} \]

\[ X = C(1 - |(H' \mod 2) - 1|) \]

\[
(R_1, G_1, B_1) = \begin{cases} 
(0,0,0) & H \text{ is undefined} \\
(C, X, 0) & 0 \leq H' < 1 \\
(X, C, 0) & 1 \leq H' < 2 \\
(0, C, X) & 2 \leq H' < 3 \\
(0, X, C) & 3 \leq H' < 4 \\
(X, 0, C) & 4 \leq H' < 5 \\
(C, 0, X) & 5 \leq H' < 6 
\end{cases} \]

\[ m = V - C \]

\[ (R, G, B) = (R_1 + m, G_1 + m, B_1 + m) \]
RGBA

- In practice, it is very common to add another component, the **alpha** component.
- The alpha component is the **opacity** of a channel.
- If alpha is all the way on (1 or 255), the color is opaque.
- If alpha is all the way off (0), the color is transparent.
- For final image data, alpha is usually unimportant, but it is crucial if you are going to draw images on top of other images.
- Also, 8 bits per component \( \times 4 = 32 \) bits.
- Because we have 32 and 64 bit machines, representing colors as RGBA is convenient.
We will be thinking of images as linear buffers of data (which will usually store R, G, B and sometimes A values for each pixel)

- Bitmaps (.bmp files) are almost that simple
- Most common image formats (.jpg, .png, and .gif files) are more complex
- They use different forms of compression to keep the image size small
- Otherwise, an 800 x 600 image is 3 bytes per pixel x 800 x 600 = 1,440,000 bytes > 1 MB
JPEG or JPG images

- Stands for Joint Photographic Experts Group
- Good for images without too much high contrast (sharp edges)
- Photographs are often stored as JPEGs
- Uses crazy math (discrete cosine transform) to reduce the amount of data needed
- Lossy compression
PNG images

- Good for images with low numbers of colors and high contrast differences
- Has built-in compression sort of like zip files
- Similar to the older GIF (.gif) images
  - GIFs are unpopular now because they only support 256 colors
  - GIFs also suffered from legal battles over the algorithm used for compression
- **Lossless** compression
Upcoming
Next time...

- C# and MonoGame overview
No required reading ahead of time
But, checking out MonoGame is not a bad idea
To run it on your own machine, you'll need:
- Visual Studio 2010 or higher
  - You should be able to get a full free academic version from Dr. Leap or you could use the free Visual Studio Express
- Latest DirectX runtime