CS 1124
Media Computation

Steve Harrison
August 25, 2008

WELCOME

- **Professor**
  - Steve Harrison
    - srh@vt.edu
    - main office in CRC: 1121 VTKW II, by appointment
    - office on campus: 638 McBryde Monday 9:30-11:00, Tuesday 12:30-2:00

- **Graduate Teaching Assistant**
  - Matt Schaefer
    - mschaefer@vt.edu <-- weekdays ONLY!
    - office in CRC, by appointment
    - office hours: Tuesdays 3:30-5:00 in 102 McBryde

- **backup GTA**
  - Bobby Beaton
    - rbeaton@vt.edu
    - office in CRC, by appointment

- **Purpose**
  - New course created to provide path for CS majors/minors who are interested in doing creative things with computers -- WE EVEN HAVE A TRACK CALLED CREATIVE COMPUTING!
  - also intro for students in other creative computing areas like art and music
  - Based on course at Georgia Tech
  - Goal is to teach computation in ways to attract new students to Computer Science

OR ...

if you think computers are cool, then lets
do some cool things with them.

Cool like this...

Cool stuff like this ...

We will take images, and do manipulations with code.
For example:

```python
def clearRed(picture):
  for pixel in getPixels(picture):
    setRed(pixel,0)
```

```python
def greyscale(picture):
  for p in getPixels(picture):
    redness=getRed(p)
    greenness=getGreen(p)
    blueness=getBlue(p)
    luminance=(redness+blueness+greenness)/3
    setColor(p,
 makeColor(luminance,luminance,luminance))
```

```python
def negative(picture):
  for px in getPixels(picture):
    red=getRed(px)
    green=getGreen(px)
    blue=getBlue(px)
    negColor=makeColor(255-red,255-green,255-blue)
    setColor(px,negColor)
```

And another example ..


```
def chromakey2(source, bg):
    for p in getPixels(source):
        if (getRed(p) + getGreen(p) < getBlue(p)):
            setColor(p, getColor(getPixel(bg, getX(p), getY(p))))
    return source
```

Course Objectives

- Students will be able to read, understand, modify, and assemble from pieces programs that achieve useful communication tasks: Image manipulation, sound synthesis and editing, text (e.g., HTML) creation and manipulation, and digital video effects.

- Students will learn what computer science is about, especially data representations, algorithms, encodings, forms of programming.

- Students will learn useful computing skills

Who should take this course?

- Any student who wants to use computers for creative purposes

- CS majors

  - alternative to Intro to Java (CS 1704)

  - to go on, will need Math 1205 or Math 1506, and Engineering Ed 1024

You should know ...

- Website http://moodle.cs.vt.edu/
  - login with your pid and go to the course website
  - Announcements (You are responsible for these! Check daily!!)
  - Syllabus
  - Homeworks
  - TA Information
  - Discussion forums
  - Slides
  - Podcasts (maybe even loan you an iPod for the semester!)
- Office hours for you to come and get help
- Classroom activites include lectures + playtime with JES, so bring your laptop

Textbook
(& software source)
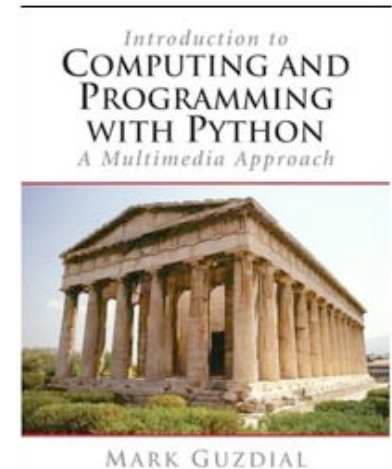
- Book from Pearson/Prentice Hall:
  "Introduction to Computing and Programming in Python:
  A Multimedia Approach" by Mark Guzdial, 2005.
- Be sure to get 2005 edition !!!!!
- CD with software and media attached to book

Organization of class

- "Lectures" Monday & Wednesday
  - here (316 McBryde)
  - lectures
  - problem solving
  - examples
  - discussion
  - practice
- "Lab" on Fridays
  - lab room (1080 Torgersen)
  - complete exercises and homework help
  - special class projects (learn about design!)

Organization of class

- "Lectures" Monday & Wednesday
  - this week:
    - Monday - introduction
    - Wednesday - introduction to Python
    - on your own:
      - read chapters 1 & 2
      - take quiz on website, due 2:00 PM Friday
- "Lab" on Fridays
  - this week:
    - download current version of JES (python)

You gotta come to class...

- My approach to the class is apprenticeship
- I'll demonstrate a lot of what you need to do
- You will see me make mistakes.
- You will also see how to fix them (with your help and that of the GTA)

- *And you will learn from each other*

- Class attendance is expected and required
- If you miss class, you are still responsible for everything that was covered.

Grading

- Online quizzes - 20%
- homework programming assignments - 45%
  - 6 or 7 individual assignments
- group programming assignments - 10%
  - 3 assignments
  - Creativity will count!
- Two in-class midterms - 10% (5% each)
- Final exam - 15%
  - extra credit will count towards final
- No curve, but ...

Improvement Policy

- Weekly Quiz: drop lowest score

- Improve one whole grade
  - Programming Projects: from previous, then previous = (previous+current) / 2
  - Midterms: second from first, then first = (first + second) / 2
  - Final: final from average of both midterms, then both midterms = (first + second + final) / 3

Hints...

- Do the examples!

- Try them out for yourself. Replicate them.
  Alter them, see what happens.
  Understand them!
- For every class, type in at least 2 of the examples from class
- Try to figure out why each line is there.

... and come talk to us

- Talk to the gTA

- Talk to me

- use the discussion board on moodle

- tell us what is working and what isn't!

Have fun

- In fact, having fun with computer programming is as important to learning as memorizing a bunch of terms. (Maybe more.)

Questions

Survey

- Please complete this survey before leaving!

- If you are wait listed, please take the survey. (We'll see how many can be added on Wednesday).

Course ID = First 3 digits + middle 3 digits
+ last 3 digits of VT ID

If VT ID = "123456789" then
     123
     456
+ 789

   1368

Coming attractions

- For Thursday:
  - Read chapters 1 & 2
- For Friday:
  - Online Quiz 1 due on Friday at 2:00 PM (on the reading)
- For Monday
  - Chapter 3
  - Online Quiz 2 due 9:00 AM (on the reading)