Final project

Mon., 4/22 – hw11 + project start due
Mon., 4/29 – hw12 + milestone due
Fri., 5/3 – 8pm: final proj. due

Mon., 4/22 – hw11 + project start due
Mon., 4/29 – hw12 + milestone due
Fri., 5/3 – 8pm: final proj. due

Final-project preview!

Choices of final project:

TextID
Picobot
vPython
Connect 4 AI

Today ~ overview

next time ~ overview

---

Final 2 lab days...

Labs will meet the last 2 weeks of class, but...

- they're "extra-optional" ~ no lab prob.
- labtime: final projects ~ assistance + progress
- there are theocomp problems, too (hw12)
- Plus, hw12 has up to +50 pts of extra credit!

'tis the season for final projects...

Today is about the CS 5 final projects + it's a sales pitch for the four possible options:

Picobot
C4 AI
TextID
vPool

I’ve got my eyes on some of these projects!

Eye’ll bet!
Final projects

Final CS hw

open-ended
comprehensive
same projects across sections
several choices...

Working in teams of 1-3 is OK
Teams need to work together - in the same place -
and need to share the work equally...
Pairs/trios are welcome (larger should split)
Teaming is extra-encouraged on the final project!

4/22
- "Start" ~ part of hw11

When?!

Mon. 4/29
- "Milestone" ~ part of hw12
- project-specific tasks to help w/progress...

Fri. 5/3
- Final project & short reflection on how to run it and how it went.
- due at 8pm
- Euros ok; grutoring tapers.

The projects...

vPool: 3D graphics
Connect Four: Game AI
TextID: Text-style matching
Picobot Project: Genetic algorithms

Start, milestone, and final-project submission

Dates:
- The final version is due Friday, May 3rd at 8pm
- The milestone version is due Monday, April 29th by 11:59pm.
- The starter version is due Monday, April 22nd by 11:59pm.
3d graphics-based game using VPython

Physics engine...

... it's not really very constrained at all!

A few constraints...
- need ≥4 physically interacting objects
- allow the user to direct 1+ objects, either by keyboard or mouse or both
- needs a game goal + be winnable!
- must detect some "linear" and some "spherical" collisions and implement their results on the motion of the objects

Example vProjects...

The Picobot project

Big idea
1. Implement Picobot in Python
2. Train Python to write successful Picobot programs!

Picobot, behind the curtain...

What data structures (classes) might be helpful in implementing Picobot?

Talk about going full circle...
Picobot's classes

```python
class Program:

    What type should self.rules be?

0 xxxx -> N 0
0 Nxxx -> W 0
0 NxWx -> S 0
0 xxWx -> E 0
0 xxxS -> E 0
0 xExS -> N 0
0 xExx -> N 0
0 NExx -> S 1
1 xxxx -> S 1
1 Nxxx -> E 1
1 NxWx -> E 1
1 xxWx -> N 1
1 xxWS -> N 1
1 xxxS -> W 1
1 xExS -> W 1
1 xExx -> S 1
1 NExx -> W 0
```

Picobot's classes

```python
class World:

    What type in Python could most usefully hold the environment?

    What class that you've already written will be most similar to Picobot's World?

    What will self.room be?
```

Your actual ASCII is likely to be more monochromatic!

http://rednuht.org/genetic_cars_2/ or http://boxcar2d.com/
Box2d: https://www.youtube.com/watch?v=uxourrlPlf8

Program evolution

An example of genetic algorithms, which are used for optimizing hard-to-describe functions with easily-splittable solutions.

Start with a population of, say, ~200 random Picobot programs...
Repeat this "survival of the fittest" process for many generations...

... and by the end, your Python code should/will have evolved a much more capable Picobot program!

What the goal?

What CSers "do"

final project algorithms...
A couple of years ago...

Though Robin Ellacott’s twenty-five years of life had seen their moments of drama and incident, she had never before woken up in the certain knowledge that she would remember the coming day for as long as she lived.
**Quiz**

Algorithmic Intuition...

Dictionary-comparing

Suppose these two Python dictionaries count words from various texts, e.g., by J.K. Rowling and W. Shakespeare. Which of these two text models does the third dictionary, at right – with unknown author -- better match? Why?

<table>
<thead>
<tr>
<th>JKR</th>
<th>WS</th>
</tr>
</thead>
</table>
| { "love": 25,  
  "spell": 275, 
  "thou": 42 } | { "love": 3,  
  "thou": 1,  
  "potter": 2,  
  "spam": 4 } |

word-count dictionary
for an unknown author

does this better match
JKR or WS? Why?

Extra: what algorithm would you devise to quantify the two matches here?!?

Naïve Bayes classification

Bayesian spam filtering

From Wikipedia, the free encyclopedia

Bayesian spam filtering (IPA: /ˈbeɪziən ənˈzɛm ər, after Rev. Thomas Bayes) is a statistical technique of e-mail filtering. In its basic form, it makes use of a naïve Bayes classifier on bag of words features to identify spam e-mail, an approach commonly used in text classification.

Model matching

Suppose we have two trained models:

<table>
<thead>
<tr>
<th>WS</th>
<th>JKR</th>
</tr>
</thead>
</table>
| { "love": 50,  
  "spell": 8,  
  "thou": 42 } | { "love": 25,  
  "spell": 275,  
  "thou": 42 } |
| { "love": 25,  
  "spell": 275,  
  "potter": 700 } | { "love": 25,  
  "spell": 275,  
  "potter": 700 } |

how do we handle
different-sized texts?

Unknown text: { "love": 3,  
  "thou": 1,  
  "potter": 2,  
  "spam": 4 }  

normalize for size

<table>
<thead>
<tr>
<th>WS</th>
<th>JKR</th>
</tr>
</thead>
</table>
| { "love": 0.50,  
  "spell": 0.08,  
  "thou": 0.42 } | { "love": 0.025,  
  "spell": 0.275,  
  "potter": 0.700 } |

normalize for size

| Unknown text: { "love": 3,  
  "thou": 1,  
  "potter": 2,  
  "spam": 4 } |

These must have been some really avant-garde texts!
Dictionary-comparing

Suppose these two Python dictionaries count words from various texts, e.g., by J.K. Rowling and W. Shakespeare. Which of these two text models does the third dictionary, at right – with unknown author -- better match? Why?

**JKR**

```
{ "love": 25,
  "spell": 275,
  "potter": 700 }
```

word-count dictionary for J.K. Rowling

**WS**

```
{ "love": 50,
  "spell": 8,
  "thou": 42 }
```

word-count dictionary for W. Shakespeare

```
{ "love": 3,
  "thou": 1,
  "potter": 2,
  "spam": 4 }
```

does this better match JKR or WS? Why?

**Extra:** what algorithm would you devise to quantify the two matches here?!?