

cis1.5
introduction to computing using c++
(robotics applications)
spring 2009
lecture # 1.1
introduction

topics:

- introduction to the course
- what is a computer?

instructor:

- Prof Elizabeth Sklar, sklar@sci.brooklyn.cuny.edu, AIM screen name: *agentprof*

course web page:

- <http://www.sci.brooklyn.cuny.edu/~sklar/cis1.5>

introduction to the course

• my pitch:

- This course is ultimately about **control**!!!
- Learn how to control computers and robots and a surprisingly large number of devices and other seemingly non-technical components that you encounter in your everyday life!
- Today, technology is ubiquitous—learn how to control it before it takes control of you!

• but seriously:

- introduction to computer programming using the C++ language
- uses **robotics** as a *context* (i.e., the basis for examples and some of the lab exercises)
- not open to students who are enrolled in or have completed CIS 1.10 or 1.20 or 2.80 or 15 or 16

topics

• the following topics will be covered in 6 units:

- I. displaying simple information and remembering it
(*output and data*)
- II. reading simple information and making decisions about it
(*input and control structures*)
- III. behaving efficiently
(*functions*)
- IV. dealing with complex information
(*arrays and strings*)
- V. doing interesting things with all kinds of information
(*searching and sorting*)
- VI. organizing programs
(*simple classes*)

course structure

• 6 units

• each unit has:

- 1-3 **lectures**
- 2-3 **labs**
- 1 **assessment**

• the labs will be hands-on sessions using laptops in the classroom

• the assessments will be:

- programming projects

• your grade = 6 projects (10% each) + midterm (10%) + final exam (30%)

special section



- newly developed course materials, hardware and software will be used
- including the cool and funky Scribbler robot
- development and implementation of the materials for this course are partially sponsored by a grant from IPRE, the Institute for Personal Robots in Education:

<http://www.roboteducation.org>

- a unique opportunity to try newly developed materials and provide feedback!

about me

- undergrad: Barnard, CS major/English minor, BA 1985
- 10 years of industry experience working as a scientific and business programmer
 - MIT Lincoln Laboratory (<http://www.ll.mit.edu/>)
- grad school: Brandeis University, MA 1997, PhD 2000
- previous teaching:
 - Monash University, Melbourne, Australia (Summer 2000)
 - University of Melbourne, Melbourne, Australia (Summer 2000)
 - Boston College, Massachusetts (Fall 2000–Spring 2001)
 - Columbia University (Fall 2001–Spring 2005)
 - Brooklyn College, Fall 2005– ...
- research interests center around educational technologies:
 - multi-media learning environments and educational robotics
 - interactive learning systems
 - multi-agent simulation and artificial intelligence (AI)

how to learn a programming language

- YOU are responsible for your own learning!!!
- I will point you in the right direction...
- but YOU must PRACTICE, PRACTICE, PRACTICE...
- and PRACTICE some more!!!
- It's like learning to play the piano—you have to put your fingers on the keyboard!
- If you don't understand, then ASK for help!
- You need to be ready to THINK and learn how to figure things out.
- Think "outside the box"!

getting started

- programming is like solving puzzles
- think differently
- the world is now made up of

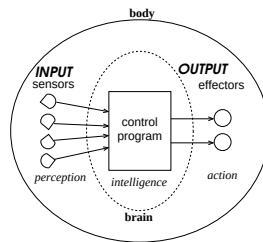
objects
and
actions

or

agents
and
behaviors

what is an agent?

- an *autonomous* entity that exists and interacts in an *environment*
- contains *sensors* to perceive its own state and its surrounding environment
- possesses *effectors* (e.g., motors) which perform actions or *behaviors*
- has a *control program* which takes input from the sensors, makes *intelligent* decisions about actions to take, and outputs those actions by sending commands to effectors
- a *robot* is an *embodied agent* that interacts in a *physical environment*



what is a control program?

- a *control program* is a set of instructions that tells the agent what to do
- a *programmer* is a person who writes those instructions
- there are many different *programming languages* that one can use to write programs—in this class, we will learn **C++**
- C++ is called a *high-level language* because:
 - it is kind of like English (no, really!)
 - well, it is more like English than the *low-level machine language* that the computer understands
- a *compiler* will translate a program from a high-level language into low-level machine language
- then the agent will *execute* those commands by reading the instructions in the low-level machine language that it understands
- programming an agent is the same thing as programming a computer

commands

- an agent follows commands
 - commands = series of instructions*
- you will learn how to *command* an agent
 - command = program = write instructions*
- you understand the commands, but does the agent?
 - that's a question of cognition...
 - Artificial Intelligence, Cognitive Science
- what do you think?

to do

- in class:
 - fill out pre-semester survey and give it to me before you leave today
 - work on today's lab
- at home:
 - check out the class web page:
<http://www.sci.brooklyn.cuny.edu/~sklar/cis1.5>