cc3.12/cis1.0
computing: nature, power and limits—robotics applications
fall 2007
lecture # I.1
introduction

topics:

(0) introduction to the course

(1) what is a computer?

(2) what is a robot?

(3) to do

instructor:

- Prof Elizabeth Sklar, *sklar@sci.brooklyn.cuny.edu*

course web page:

- `http://www.sci.brooklyn.cuny.edu/~sklar/cc3.12`

(0) introduction to the course

- about this course
  - uses **robotics** as a *context* (i.e., the basis for examples and some of the lab exercises)
- topics covered:

(I) Introduction to Computers and Networks
(II) Algorithms and Computer Languages
(III) Data representation and Storage
(IV) Event-driven Programming
(V) Control Structures
(VI) Solvability and Feasibility
(VII) Security, Privacy, Encryption and Plagiarism

(0) course structure

- 7 **units**
- each unit has:
  - 1-2 **lectures**
  - 1-2 **labs**
  - 1 **assignment**
- some of the labs will be hands-on sessions using the internet in a multimedia classroom (room 106 F, in the WEB building)
- the rest of the labs will be hands-on sessions using **LEGO Mindstorms robots** (in 4411 N)
- your grade =
  - 7 assignments (53% total)
  - attendance (7%)
  - midterm (10%)
  - final (30%)

(1) what is a computer?

- a device that can process data, store data and execute instructions
  - what is the difference between a computer and a calculator?
  - what everyday devices have computers inside them?

(1) hardware components of a computer

- processor (i.e., central processing unit, or CPU)
- memory
  - short-term: RAM (random access memory), goes away when you turn off the computer
  - long-term: permanent storage media, like a hard disk, USB drive, CD
- input devices
- output devices
- peripherals




(1) software components of a computer

- operating system
  - Microsoft Windows
  - Mac OS-X
  - Linux
  - UNIX
- applications
  - email (Outlook, MacMail, Eudora, pine, ...)
  - browser (Firefox, Internet Explorer, Safari, ...)
  - music (iTunes)
  - office tools (word processing, spreadsheets, presentations)
  - calculator
- drivers
  - printer
  - scanner

(1) how does software work?

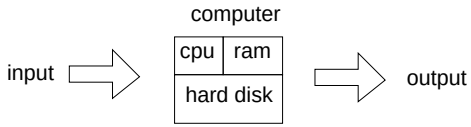
- a human writes instructions for the computer in a language that the computer can understand
  - low-level languages (e.g., assembly)
  - high-level languages (e.g., Java)
- high-level languages are **compiled** (translated) into **binary machine code**, i.e., a language that the computer's processor can understand
- instructions must be very specific!
- instructions are grouped into **programs**
- instructions are executed **sequentially** (one after another)
- what can go wrong?
  - user or "operator" errors
  - program errors: called **bugs**
  - hardware errors (or "faults")

(1) how are programs written?

- programs are written in high-level languages using a **text editor** (e.g., NotePad or TextEdit)
- this is different from a **word processor** (e.g., Microsoft Word), which stores extra formatting characters (besides what you see on the screen...)
- the **programmer** invokes a **compiler** to translate the program into code that the computer can execute
- the **user** runs the executable program
- the programmer's code and the executable program are stored on the computer's hard disk in **files**

(1) what is a computer again?

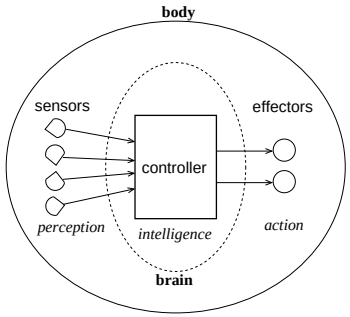
- a device that receives **input** from a human or another computer or another device, **processes** that input and produces **output**



- a computer **program** is what does the processing
- the program is stored on the computer's hard disk, and when the program runs, it is copied into the computer's memory (RAM) and the instructions contained in the program are executed by the computer's central processing unit (CPU)
  - it's like reading a book... you get the book from the shelf where it is stored (which is like fetching the program from the computer's hard drive), you open the book (which is like starting the program) and you read it, one word at a time (which is like running the program, one instruction at a time)

(2) what is a robot?

- *robot = autonomous embodied agent*
- has a *body* and a *brain* (a COMPUTER!)
- exists in the physical world (rather than the virtual or simulated world)
- is a mechanical device
- contains *sensors* to perceive its own state
- contains *sensors* to perceive its surrounding environment
- possesses *effectors* which perform actions
- has a *controller* which takes input from the sensors, makes *intelligent* decisions about actions to take, and effects those actions by sending commands to motors

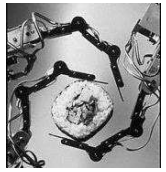


(2) a bit of robot history

- the word *robot* came from the Czech word *robota*, which means *slave*
- used first by playwrite Karel Capek, "Rossum's Universal Robots" (1923)
- human-like automated devices date as far back as ancient Greece
- modern view of a robot stems from science fiction literature
- foremost author: Isaac Asimov, "I, Robot" (1950)
- the *Three Laws of Robotics*
  1. A robot may not injure a human being, or, through inaction, allow a human being to come to harm.
  2. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
  3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.
- Hollywood broke these rules: e.g., "The Terminator" (1984)

(2) effectors

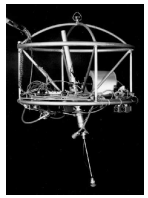
- comprises all the mechanisms through which a robot can *effect* changes on itself or its environment
- *actuator* = the actual mechanism that enables the effector to execute an action; converts software commands into physical motion
- types:
  - arm
  - leg
  - wheel
  - gripper
- categories:
  - *manipulator*
  - *mobile*



some manipulator robots

(2) mobile robots

- classified by manner of locomotion:
  - *wheeled*
  - *legged*
- stability is important
  - *static stability*
  - *dynamic stability*

(2) sensors

- convert physical properties into electronic signals which can be interpreted by the robot's brain (computer) in a useful way

property being sensed	type of sensor
contact	bump, switch
distance	ultrasound, radar, infra red (IR)
light level	photo cell, camera
sound level	microphone
smell	chemical
temperature	thermal
inclination	gyroscope
rotation	encoder
pressure	pressure gauge
altitude	altimeter

(2) autonomy

- automony = no remote control!
- the robot has to "think" by itself
- to be truly autonomous, a system must be able to accomplish *goals* and *solve problems*
- control architectures
  - *deliberative*
    - look-ahead; think, plan, then act
  - *reactive*
    - don't think, don't look ahead, just react!
  - *hybrid*
    - think but still act quickly
  - *behavior-based*
    - distribute thinking over acting

(2) the robots for our labs

- LEGO Mindstorms
- Hitachi h8300 microprocessor (computer) called **RCX**
- with an IR (infra-red) transceiver

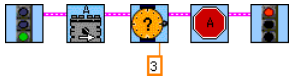
- and 3 input ports, for:
  - light sensor
  - touch sensor

- and 3 output ports, for:
  - motors
  - light bulbs

(2) programming the LEGO Mindstorms

- you write programs on a computer and *download* them to the RCX using an IR transmitter ("communication tower")

- we will use **RoboLab** — a graphical programming environment



(3) to do.

- get a copy of the textbook (A Balanced Introduction to Computer Science, by David Reed, published by Prentice-Hall, 2004)
- ... and start to read chapter 1
- check out the class web page:
  `http://www.sci.brooklyn.cuny.edu/~sklar/cc3.12`

(3) about me.

- undergrad: Barnard, CS major, class of 1985
- 10 years of industry experience working as a scientific and business programmer
- grad school: Brandeis University, PhD 2000
- previous teaching:
  - Monash University, Melbourne, Australia
  - University of Melbourne, Melbourne, Australia
  - Boston College, Massachusetts
  - Columbia University, Fall 2001–Spring 2005
  - Brooklyn College, Fall 2005– ...
- research interests center around educational technologies:
  - artificial intelligence (AI)
  - educational robotics
  - interactive learning systems
  - multiagent simulation

(3) about you.

- please take out a piece of paper and write down...
  1. your name
  2. your email address (print clearly!)
  3. your class and major OR if you are a non-matriculating student, categorize yourself
  4. your background in computers, if any
  5. why you are taking this course
  6. what you hope to learn here
  7. one sentence about one wonderful thing you did over the summer
- ...and give it to me before you leave