Overview/Questions

- Review: Programmability
- Why learn programming?
- What is a programming language?
- What is Python?
Early History of Computing

Joseph Jacquard (1801)
Jacquard’s Loom, the punched card

Review: Programmability

What tricks does your computer do?
  – Web browsing, email, instant messenger
  – Play games
  – Watch movies, organize photos
  – Word processing, spreadsheets, database

Programmability
The ability to give a general-purpose computer instructions so that it can perform new tasks.
Why Learn Programming?

– Programming …
  • is a challenging and intellectually engaging experience.
  • is an important part of computer science.
  • can be a lot of fun.
  • is a skill that can pay the bills. Very well.
– Why learn programming in CS101?

Why Learn Programming?

The fundamental ideas of CS are:
– Algorithms
– Protocols
– Encoding Information
– Abstraction

Programming teaches these ideas!
Algorithm

An algorithm is a sequence of clear and precise step-by-step instructions for solving a problem in a finite amount of time.

Programming Languages

“Computer. This is a Class-A compulsory directive. Compute, to the last digit, the value of pi.”
- Spock, on Start Trek

A program is a sequence of instructions.
Computers are not very good at understanding natural languages
– Ambiguity and imprecision
– Dialects, accents, etc
Programming Languages

Computer languages are exact and unambiguous.

Programming is the process of encoding our ideas for the computer:
- Syntax (form)
- Semantics (meaning)

Some popular programming languages:
- C++, Java, Perl, Python

Machine Language

Computer hardware performs only a small set of concrete operations, which are created in circuitry.
- Example: the “adder” circuit

Machine language instructions tell the computer which operation to perform. The instructions are provided in binary notation:

<table>
<thead>
<tr>
<th>Instruction specifier</th>
<th>0 0 0 1 0 0 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operand specifier</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0</td>
</tr>
</tbody>
</table>
What the Computer Can Do

Modern computer processors can do the following types of basic tasks (CPU instructions):
- Add, subtract, multiply, divide, increment, decrement
- Logical AND, OR, XOR, NOT, and NEG operations
- Load data in from RAM, store data out to RAM
- Load data from/send data to input/output devices
- Determine if values are equal to zero/Less than zero
- Go to another instruction (by numeric address)

Note: by data, we mean one word (e.g. 32 bits)

Partial Summary

- Even the modern computer is still a moronic number-crunching, data moving device.
- Machine language is written in binary, and has only very limited operations.

- Why would anyone want to write a program in machine language?
High-level Programming Languages

**High-level language**
A language that provides a richer (more English like) set of instructions.

Example:
```python
a = 2
b = 5
total = a + b
if total < 10:
    print total
else:
    print "Inconceivable!"
```

*How can the computer understand this kind of language?*

---

What You Learned Today

– Programmability and Programming
– Machine language
– High-level languages
Announcements and To Do List

– Download & install JES on your computer!
    (pick 4.3 -- Mac or Windows)

– Readings:
  • The CS101 Guide to Python/JES
  • Optional (free) book about Python:
    How to Think Like A Computer Scientist: Learning with Python
    Available online at: [http://openbookproject.net/thinkCSpy/](http://openbookproject.net/thinkCSpy/)

– HW9 is due Wednesday 3/30