Admin

- Next Monday and beyond, lectures meet in Cemex Aud
    - Over in GSB, no use of balcony (only main floor)
- More unix sessions today/weekend if you still need one
- Assign 0 due Monday night
- Labs start Monday

The C programming language

- **~1970, Unix, small & simple, abstractions map to hardware**
  - "C is quirky, flawed, and an enormous success"
    - — Dennis Ritchie
  - "C gives the programmer what the programmer wants; few restrictions, few complaints"
    - — Herbert Schildt
  - "C combines all the elegance and power of assembly language with all the readability and maintainability of assembly language”
    - — Unknown
- **Software/language inertia, backward compatibility**
- **The C/C++/Java "family"**
- **107 gives crash course in C for C++ programmers**
  - Focus on tricky/different parts of language
  - Use K&R, other reference to fill in gaps

C, the ancestor of Java/C++

- You will be grateful for the familiar:
  - syntax
  - data types (although no explicit Boolean type before C99)
  - arithmetic/relational/logical ops
  - control structures
- You will be sad about the conspicuous absences:
  - no syntactic niceties: overloading, default arguments, const, ref params
  - no fancy ADTs (strings, vector/list, map, etc.), no objects at all!
  - thin standard libraries (no graphics, networking, etc)
  - little help with memory management
  - weak static checking, little runtime support
- You might come to appreciate:
  - low-level control/efficiency
  - little abstraction penalty, no surprises
  - small language footprint