COMSW 1003-1

Introduction to Computer Programming in C

Lecture 1

Spring 2011

Instructor: Michele Merler

http://www1.cs.columbia.edu/~mmerler/comsw1003-1.html

Course Information - Goals

"A general introduction to computer science concepts, algorithmic problem-solving capabilities, and programming skills in C"

University bulletin

- Learn how to program, in C
- Understand basic Computer Science problems
- Learn about basic data structures
- Start to think as a computer scientist
- **Use all of the above to solve real world problems**

Course Information - Instructor

- Michele Merler
  - Email: mmerler@cs.columbia.edu or mm3233@columbia.edu
  - Office : 624 CEPSR
  - Office Hours: Friday 12pm-2pm
- 4th year PhD Student in CS Department
- Research Interests:
  - Image & Video Processing
  - Multimedia
  - Computer Vision

Course Information- TA

- TDB
  - Email: TDB@columbia.edu
  - Office : TA room
  - Office Hours: TDB

Course Information- Courseworks

We will be using Courseworks (https://courseworks.columbia.edu/) for:

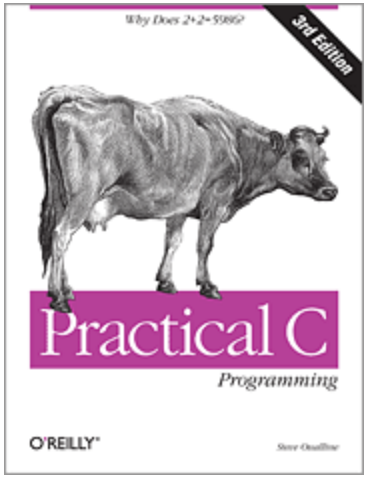
- Message board for discussions
- Submit Homeworks
- Grades

Check out the board before you send an email to the instructor or the TA, the answer you are looking for could already be there!

Course Information
Requirements and Books

Requirements

- Basic computer skills
- CUNIX account



Textbooks

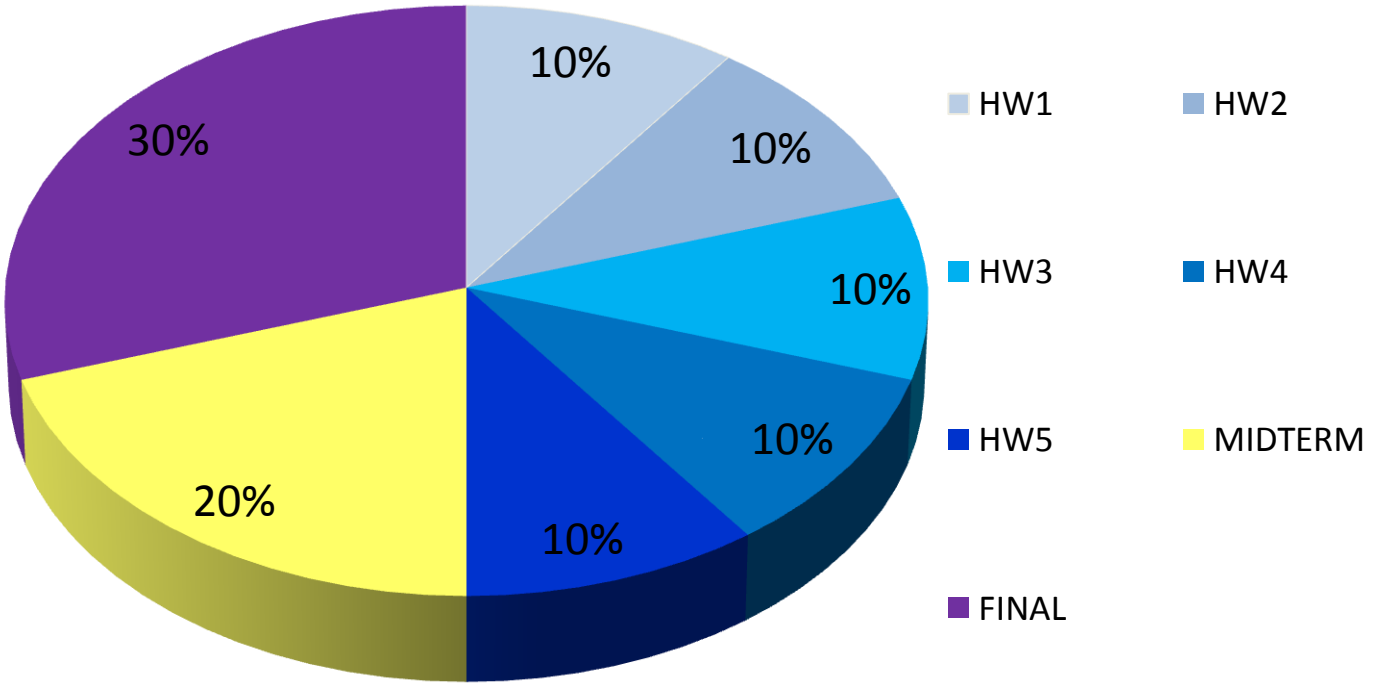
- **The C Programming Language (2nd Edition)**
  by Brian Kernighan and Dennis Ritchie

  http://www1.cs.columbia.edu/~mmerler/coms1003-1/C Programming Language.rar

- **Practical C Programming (3rd Edition)** by Steve Oualline

Course Information - Grading

- 5 Homeworks (10%, 10%, 10% , 10% , 10%)
- Midterm Exam (20%)
- Final Exam (30%)



Course Information
Academic Honesty

It's quite simple:

- Do not copy from others
- Do not let others copy from you

Do your homework **individually**

Please read through the department's policies on academic honesty
http://www.cs.columbia.edu/education/honesty/

Course Information - Syllabus

Go to class webpage

http://www1.cs.columbia.edu/~mmerler/coms1003-1_files/Syllabus.html

What is Computer Science?

Computer science (sometimes abbreviated **CS**) is the study of the theoretical foundations of **information** and **computation**, and of practical techniques for their implementation and application in computer systems

Wikipedia

"Computer science and engineering is the systematic study of algorithmic processes-their theory, analysis, design, efficiency, implementation, and application-that describe and transform information"

Comer, D. E.; Gries, D., Mulder, M. C., Tucker, A., Turner, A. J., and Young, P. R. (Jan. 1989). "Computing as a discipline". *Communications of the ACM* **32** (1): 9.

"Computer science is the study of information structures"

Wegner, P. (October 13–15, 1976). "Research paradigms in computer science". *Proceedings of the 2nd international Conference on Software Engineering*. San Francisco, California, United States

"*Computer Science is the study of all aspects of computer systems, from the theoretical foundations to the very practical aspects of managing large software projects.*"

Massey University

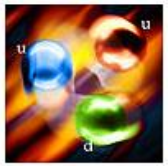
What is Computer Science?

Computer Science is the discipline that studies how to make computers perform tasks that are too complex or boring for humans

Computer Science Areas

Computational science

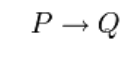

Numerical analysis

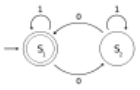

Computational physics

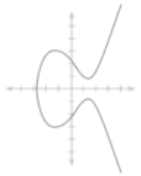

Computational chemistry


Bioinformatics

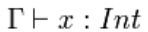
Theoretical computer science

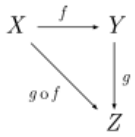

Mathematical logic


Automata theory

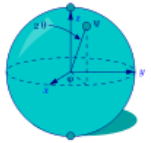

Number theory


Graph theory


Type theory

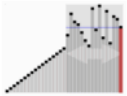

Category theory


Computational geometry


Quantum computing theory

Algorithms and data structures


Analysis of algorithms


Algorithms


Data structures

Theory of computation


Computability theory

P = NP ?

Computational complexity theory

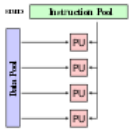
GNITIRW-TERCES

Cryptography

Computer architecture


Digital logic


Microarchitecture



Multiprocessing

Artificial Intelligence


Machine Learning


Computer vision

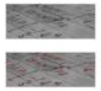
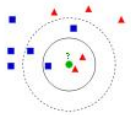
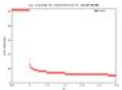


Image Processing

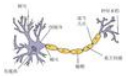

Pattern Recognition


Cognitive Science


Data Mining


Evolutionary Computation


Information Retrieval


Knowledge Representation

Natural Language Processing


Robotics


Human–computer interaction

Software Engineering

Operating systems


Computer networks


Databases


Computer security


Ubiquitous computing

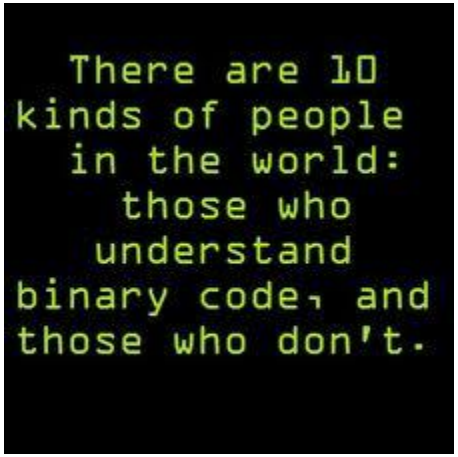

Systems architecture


Compiler design


Programming languages

Why programming?

- We need a way to tell computers what to do
- It would be nice to communicate with computers in English, but…
  - English can be ambiguous!
  - Computers only understand binary!
- Solution: programming languages



What is a Program?

- A **Program** is a sequence of instructions and computations
- We'll be designing programs in this course.
- These programs will be based on **algorithms**
- An **Algorithm** is a step-by-step problem-solving procedure

Example

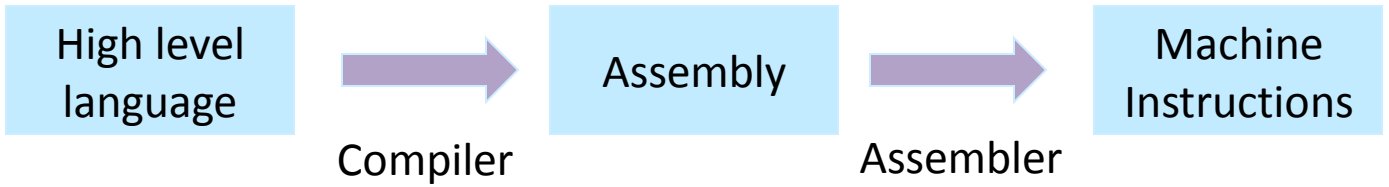
- Add 3 large numbers
  - 453 + 782 + 17,892
- Hard to do all at once
  - Solution: “divide and impera”!
  - (453 + 782) + 17,892 =
  - 1,235 + 17,892 = 19,127
- Algorithms help us divide and organize complex problems into sub-problems which are easier to solve (bottom-up approach)

Programming

- Back in the day, programmers wrote in **Assembly**, a language where each word stands for a single instruction

```
add     eax, edx
shl     eax, 2
add     eax, edx
shr     eax, 8
sub     cl, al
```

- But then they had to **hand translate** each instruction into binary!!!
- Solution: the **assembler**, a computer program to do the translation
- From then, programmers could worry only about writing assembly code
- Then they started to devise higher level languages (FORTRAN, COBOL, PASCAL, C, C++, JAVA, Perl, Python, etc.), which get translated into Assembly by **compilers** (we will use **GCC**, a C compiler for Unix)

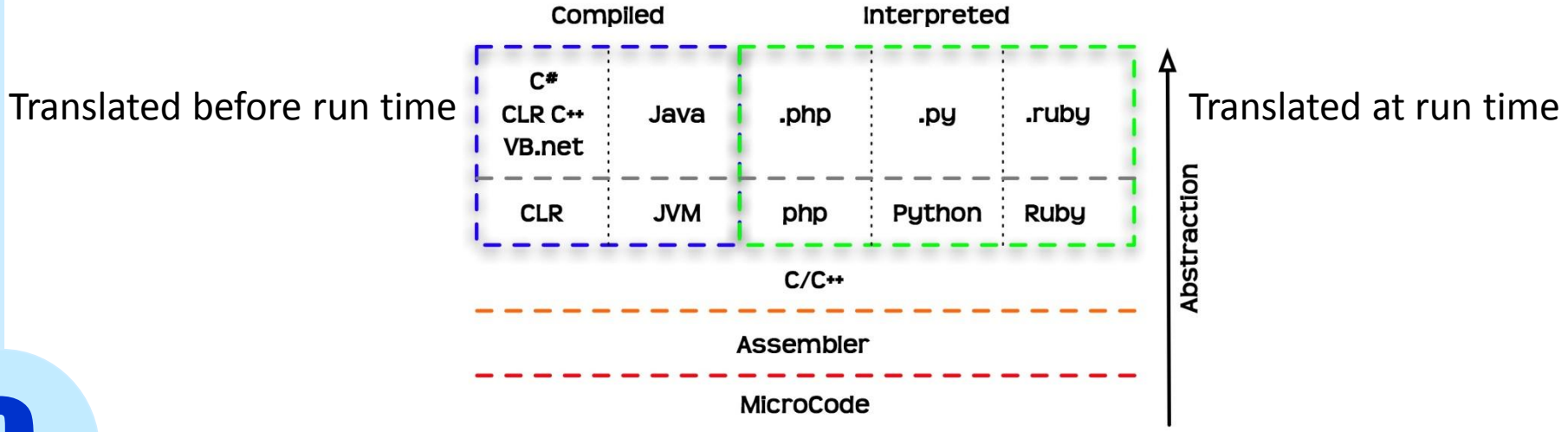


What is C?

- Programming language developed by Dennis Ritchie in 1972 at AT&T Bell labs
- Why is it named “C”?
  Well… the B programming language already existed !
- C is still the most used programming language for Operating Systems
- Popular because:
  - Flexible
  - C compiler was widely available
- Basis for other popular programming languages: C++, C#

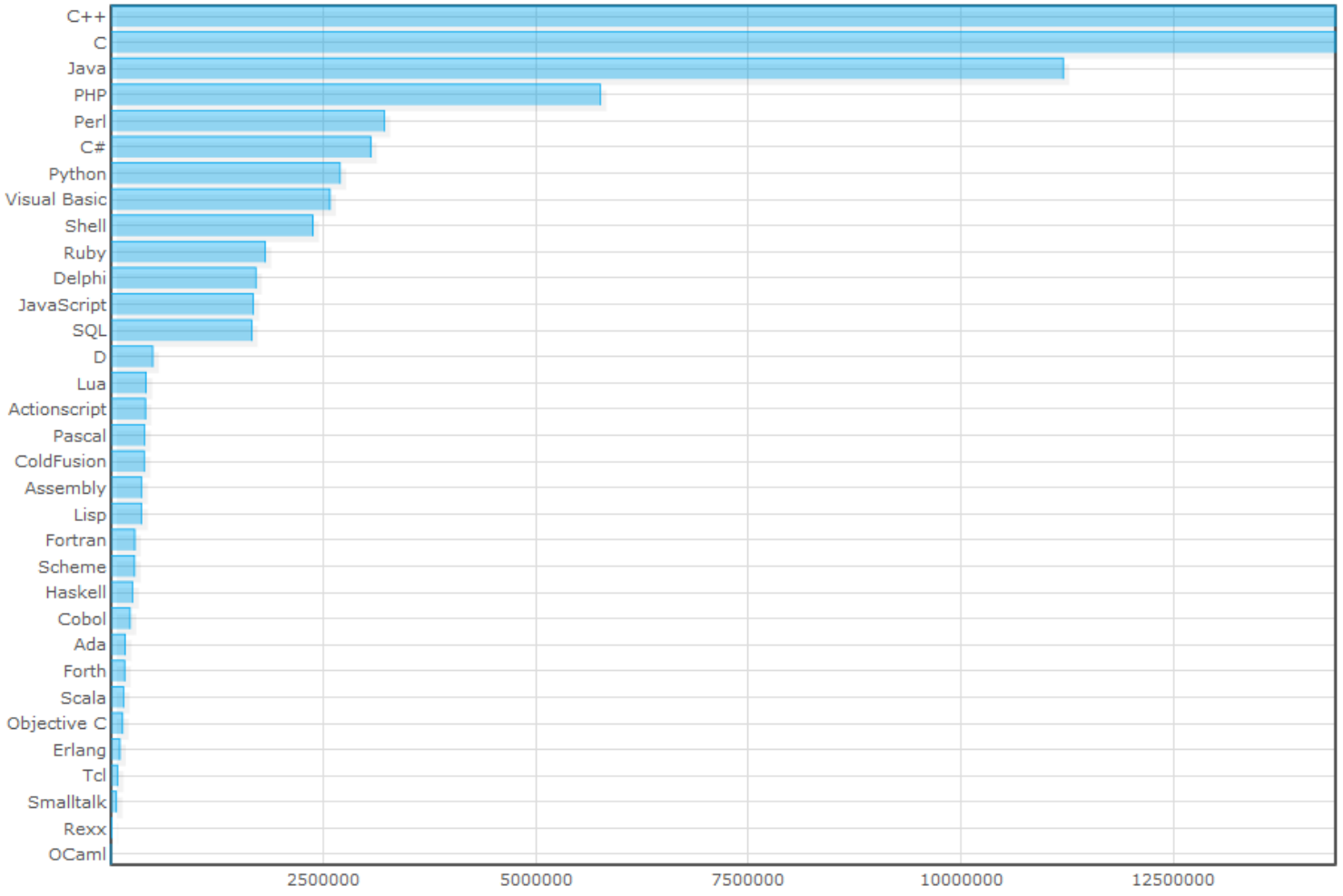
What is C?

- Among the “high level” programming languages, C is one with the lowest level of abstraction
- Close to English, but more precise!
- Easy to compile into Assembly => Fast
- Rich set of standard function = we don’t have to implement everything from scratch!



Why C? Interesting Facts …

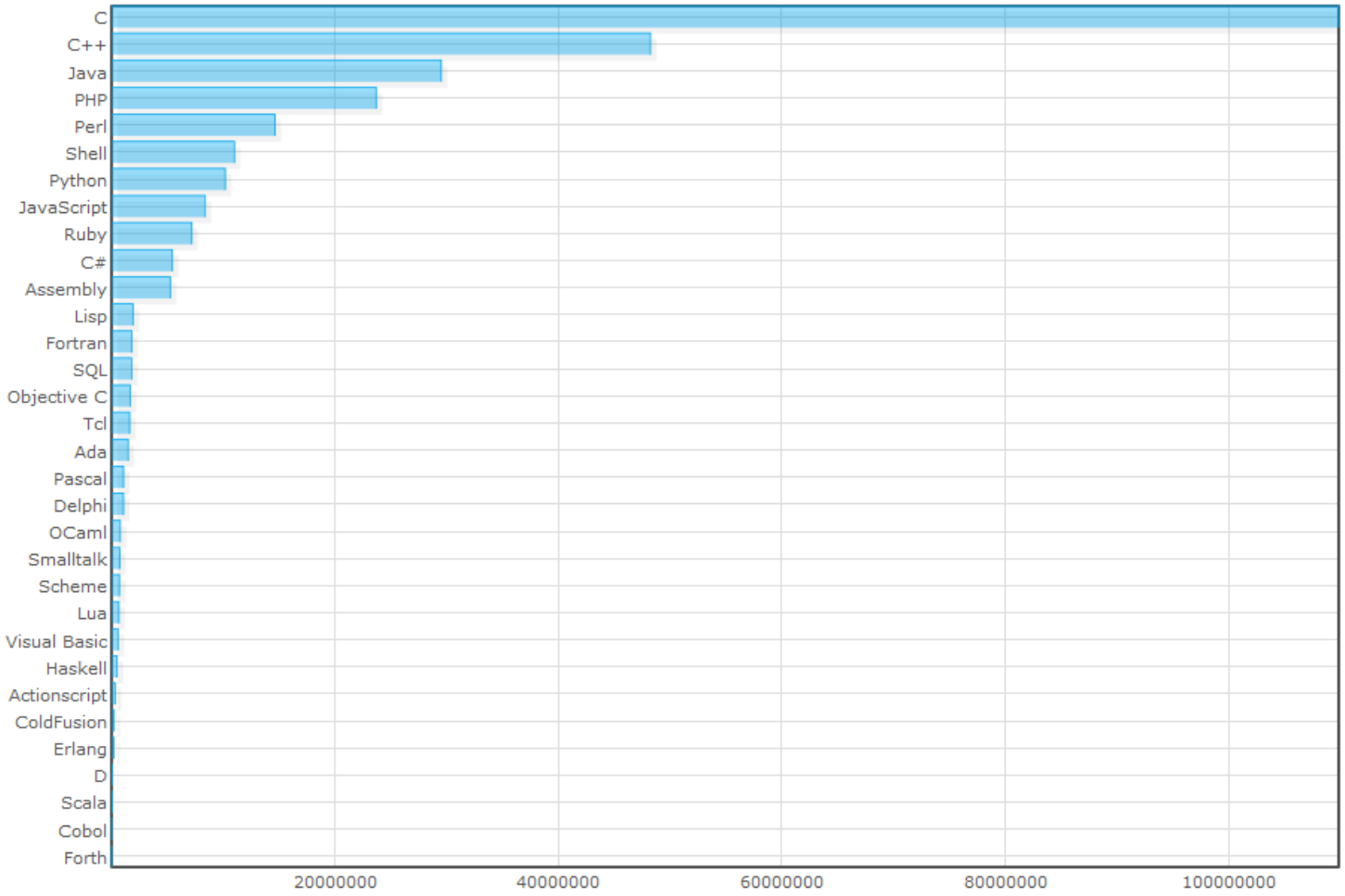
Approximation of **popularity** of language using Yahoo API http://www.langpop.com/



Slide credit: Priyank Singh

Why C? Interesting Facts …

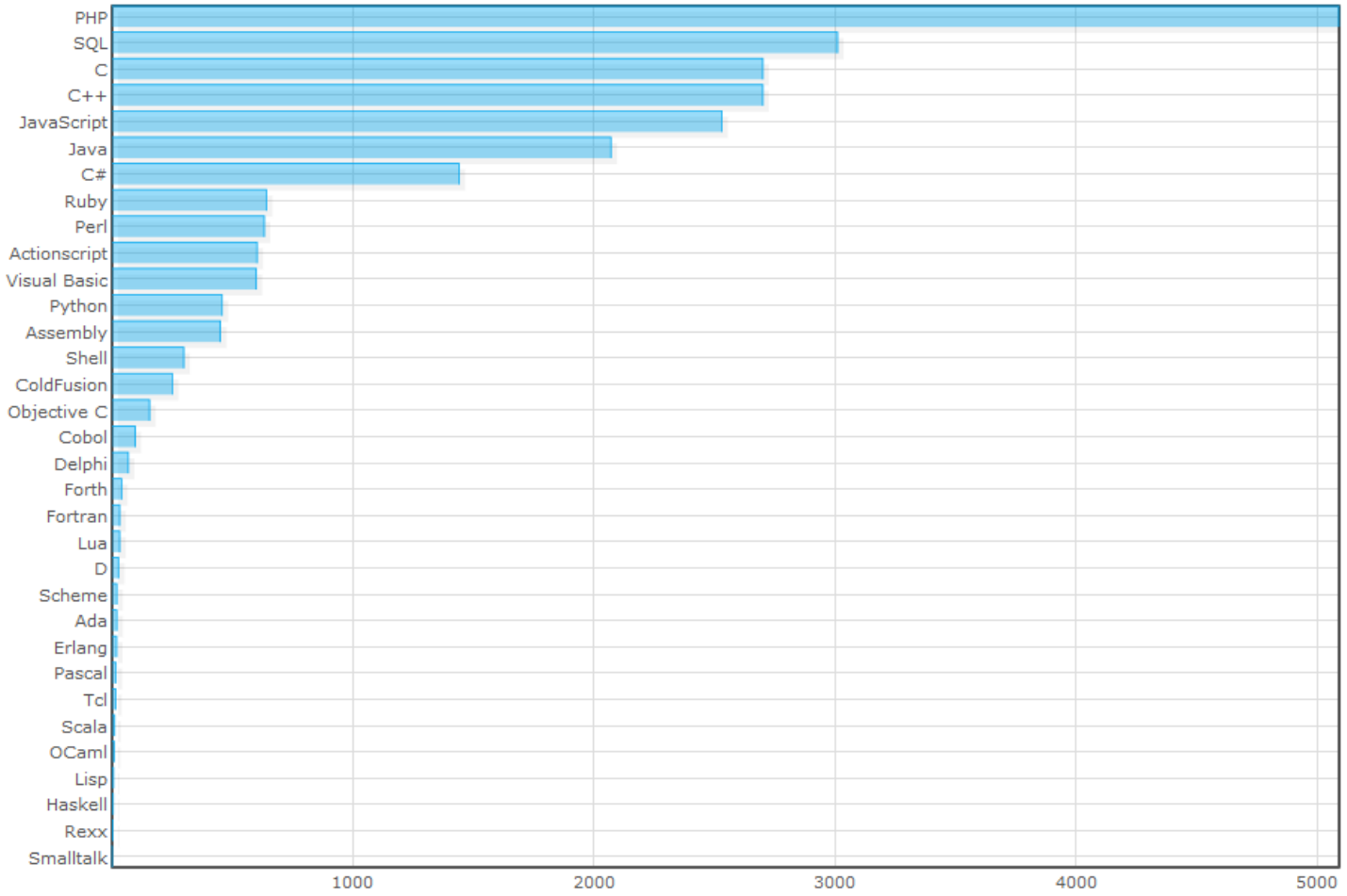
Available language **code** available using Google code search http://www.langpop.com/



Slide credit: Priyank Singh

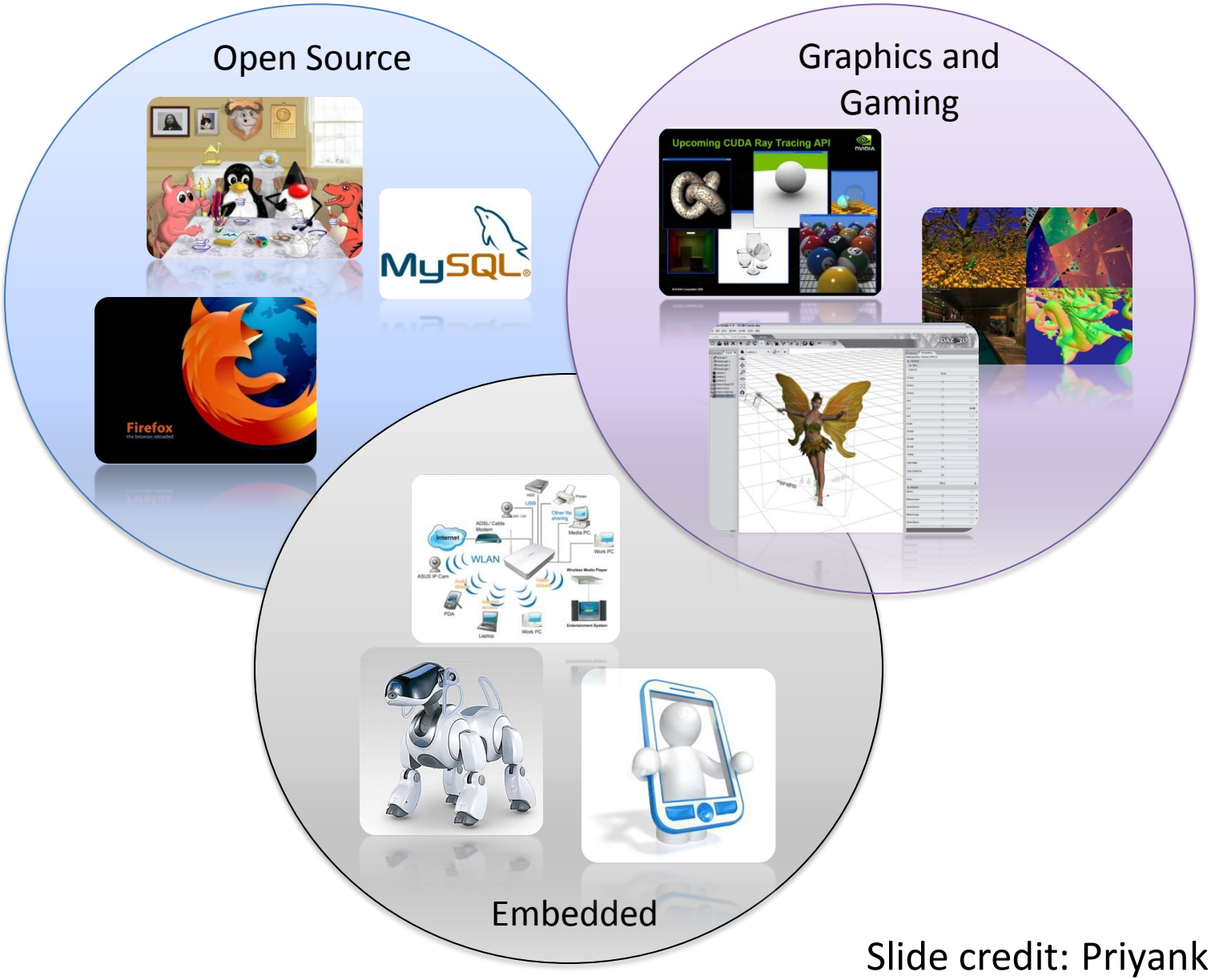
Why C? Interesting Facts …

Jobs posting on craiglist.org, from website http://www.langpop.com/



Slide credit: Priyank Singh

C/C++ Industry



Slide credit: Priyank Singh

Example of C program

Hello world!

Announcements

- Homework 0 is out! Due at the beginning of next class
- Bring your laptop to class