Introduction to LLVM

Introduction to LLVM

- LLVM: Low Level Virtual Machine
- a collection of :
  - modular and reusable compiler
  - toolchain technologies
- sub-projects of LLVM:
  - The **LLVM Core** libraries
  - Clang
  - LLDB
  - compiler-rt
  - …

How to install LLVM

- document:

  http://llvm.org/docs/GettingStarted.html

- Steps:
  - $svn co http://llvm.org/svn/llvm-project/llvm/trunk llvm

    (may need to install subversion:$sudo apt-get install subversion)
  - $cd llvm/tools
  - $svn co http://llvm.org/svn/llvm-project/cfe/trunk clang
  - $cd llvm/projects
  - $svn co http://llvm.org/svn/llvm-project/compiler-rt/trunk compiler-rt
  - $mkdir build
  - $cd build
  - ../llvm/configure
  - make

LLVM instruction

- A kind of IR (Intermediate Representation)
- Website document:
  - http://llvm.org/docs/LangRef.html
- Some useful instructions:
  - Operator:

    mul, sdiv, srem, add, sub, shl…
  - Memory operation:

    load, store…
  - Branch operation:

    icmp, br, ret…

Generate LLVM IR by clang

- Clang :an C/C++/Objective-C compiler
- used like GCC; faster than GCC
- build c source into LLVM instructions:
  - $clang -emit-llvm test.c -S -o test.ll
- run LLVM instructions:
  - $lli test.ll
- *.ll :
  - need to be generated in project 2

Generate LLVM IR(con't)

- Special functions in project2 :
  - read() -> scanf()
  - write() -> printf()
- You can compile these two origin functions by clang to see how to read and write!

Resources

- Other compiler course:
  - Stanford:

    http://web.stanford.edu/class/cs143/
  - UCB:

    http://www-inst.eecs.berkeley.edu/~cs164/archives.html
  - CMU:

    https://www.cs.cmu.edu/~fp/courses/15411-f13/