Compiler Design
编 译 器 构 造 实 验

Lab 0：实验环境及工具

教师：张献伟

助教：吴坎

xianweiz.github.io

DCS292, 2/17/2022

Linux Environment

- 所有的实验项目预期是Linux环境
  - 实现语言为C/C++
  - 需要熟悉Terminal和基本的commands，以及Vim、Emacs或其他编辑工具
  - 也可以在windows环境下完成，在提交前通过Linux环境（如WSL、docker）下的测试

- 哪些Linux环境可以使用？
  - 虚拟机
  - 本地：Debian(推荐)/Ubuntu, RedHat/CentOS
  - 远程：通过Putty, MobaXterm等连接Linux服务器
  - 在线：https://cocalc.com/doc/terminal.html

Linux Commands

- Show current directory
  - $pwd
- Change directory
  - $cd *<your_dest_dir>*
- Create a directory
  - $mkdir *<your_dir>*
- Create a file
  - $touch *<your_file>*
- Rename
  - $mv *<old_name> <new_name>*
- Remove
  - $rm [-i/r/f] *<your_dir_or_file>*

https://ubuntu.com/tutorials/command-line-for-beginners

Vim Commands

- Open a file
  - $vim *<your_file>*

- Enter insert mode
  - i

- Edit

- Exit insert mode
  - ESC

- Save and close the file
  - :wq

https://www.linux.com/training-tutorials/vim-101-beginners-guide-vim/

Let's get start ...

- Set up a folder
  - $mkdir lab0
  - $cd lab0

- Write the source code
  - $vim hello.c
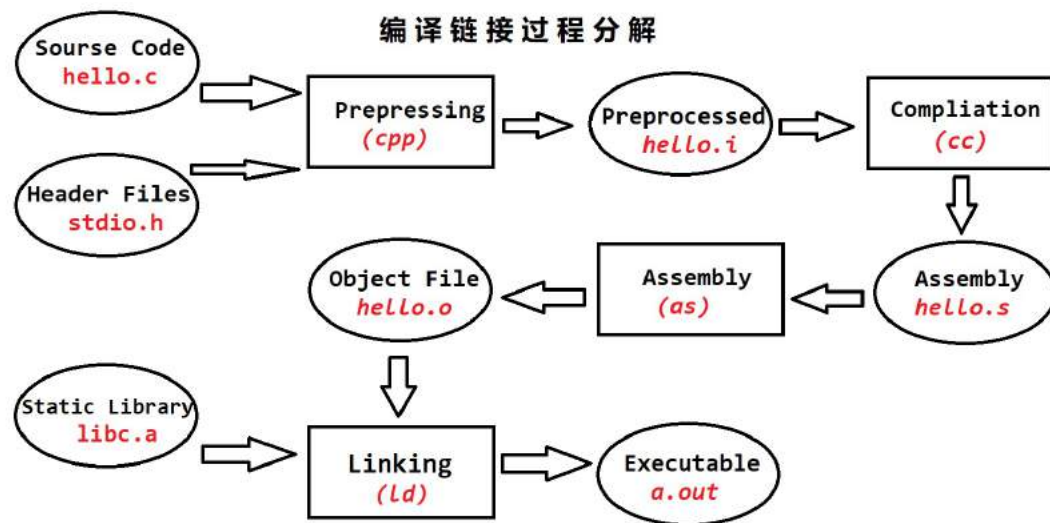
```c
#include <stdio.h>

int main()
{
    printf("Hello World!\n");

    return 0;
}
```

Build it ...

- Preprocess: *.c → *.i
  - $clang -E hello.c -o hello.i
- Compile: *.i → *.s
  - $clang -S hello.i -o hello.s
- Assembly: *.s → *.o
  - $clang -c hello.s -o hello.o
- Link: *.o → exe
  - $clang hello.o -o hello

'vim' the output file in each step

编译链接过程分解

Details: http://csapp.cs.cmu.edu/2e/ch1-preview.pdf

One More Step: Disassemble[反汇编]

- $objdump -d hello.o

- $objdump -d ./hello

- Compare the <main>:What are the differences?

```
0000000000000000 <main>:
   0:   55                      push   %rbp
   1:   48 89 e5                mov    %rsp,%rbp
   4:   48 83 ec 10             sub    $0x10,%rsp
   8:   89 7d fc                mov    %edi,-0x4(%rbp)
   b:   48 89 75 f0             mov    %rsi,-0x10(%rbp)
   f:   bf 00 00 00 00          mov    $0x0,%edi
  14:   e8 00 00 00 00          callq  19 <main+0x19>
  19:   c9                      leaveq
  1a:   c3                      retq
```

```
000000000040051d <main>:
  40051d:   55                      push   %rbp
  40051e:   48 89 e5                mov    %rsp,%rbp
  400521:   48 83 ec 10             sub    $0x10,%rsp
  400525:   89 7d fc                mov    %edi,-0x4(%rbp)
  400528:   48 89 75 f0             mov    %rsi,-0x10(%rbp)
  40052c:   bf d0 05 40 00          mov    $0x4005d0,%edi
  400531:   e8 ca fe ff ff          callq  400400 <puts@plt>
  400536:   c9                      leaveq
  400537:   c3                      retq
  400538:   0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  40053f:   00
```

Makefile

- Special format file that help build and manage the compilation automatically

- Create Makefile
  - $cd lab0/
  - $vim Makefile

- Make
  - $make

- Clean
  - $make clean

http://web.mit.edu/gnu/doc/html/make_2.html

```
CC := gcc
CCFLAGS :=
EXEC := hello

# all is the default rule
all: $(EXEC)

$(EXEC): hello.o
        $(CC) $(CCFLAGS) -o $@ $^
        @echo "Built $@ successfully"

hello.o : hello.s
        $(CC) -c $^

hello.s : hello.i
        $(CC) -S $^

hello.i : hello.c
        $(CC) -E $^ > $@

clean:
        rm -f *.i *.s *.o $(EXEC)
```

CMake

- Cmake can generate Makefiles and more modern build systems such as Ninja.

- Try:
  - $git clone https://github.com/arcsysu/SYsU-lang.git
  - $cd SYsU-lang
  - $cmake -B ../sysu/build \
    -DCMAKE_INSTALL_PREFIX=../sysu \
    -DCMAKE_C_COMPILER=clang \
    -DCMAKE_CXX_COMPILER=clang++
  - $make install -j -C ../sysu/build

https://cmake.org/cmake/help/latest/index.html

Archive[归档]

- Uncompressed tar
  - Create: $tar cvf *<archive_name>*.tar *<dirname>*
  - Extract: $tar xvf *<archive_name>*.tar
  - E.g.,: $tar cvf lab0.tar lab0/

- Compressed tar ball
  - Create: $tar cvfz *<archive_name>*.tar.gz *<dirname>*
  - Extract: $tar xvfz *<archive_name>*.tar.gz
  - E.g.,: $tar cvfz lab0.tar.gz lab0/

https://www.geeksforgeeks.org/tar-command-linux-examples/

One More thing

- 2022年全国大学生编译系统设计赛
  - 预计五月份开赛
  - 与其他高校交流的好机会，含金量极高
  - 本学期实验框架与比赛要求非常接近
  - 感兴趣的同学建议提前完成实验

https://compiler.educg.net/