CS35L Software Construction Laboratory

Lab 5: Sneha Shankar

Week 3; Lecture 1

Lab Assignment 2

- hwnwdseng.htm -> buildwords -> hwords
- Buildwords
  - Read from STDIN and perform work on input
- Store the output in hwords
  - E.g. cat hwnwdseng.htm | sh buildwords > hwords

Lab Assignment 2 contd…

- How to construct buildwords?
  - Extract lines which contain words (both English and Hawaiian) (Hint: <td> tag)
  - Get lines with Hawaiian words
    - Even numbered lines
  - sed 's/<[^>]*>//g' a.html to remove all HTML tags
  - Remove leading space
    - sed 's/^\s*//g'
  - Substitute/split comma and space in between words to newline
  - Delete entries which have any character other than Hawaiian
  - Sort unique

Modifying and Rewriting Software

How to Install Software

- Linux
  - rpm (Redhat Package Management)
    - RedHat Linux (.rpm)
  - apt-get (Advanced Package Tool)
    - Debian Linux, Ubuntu Linux (.deb)
  - **Good old build process**
    - **configure, make, make install**

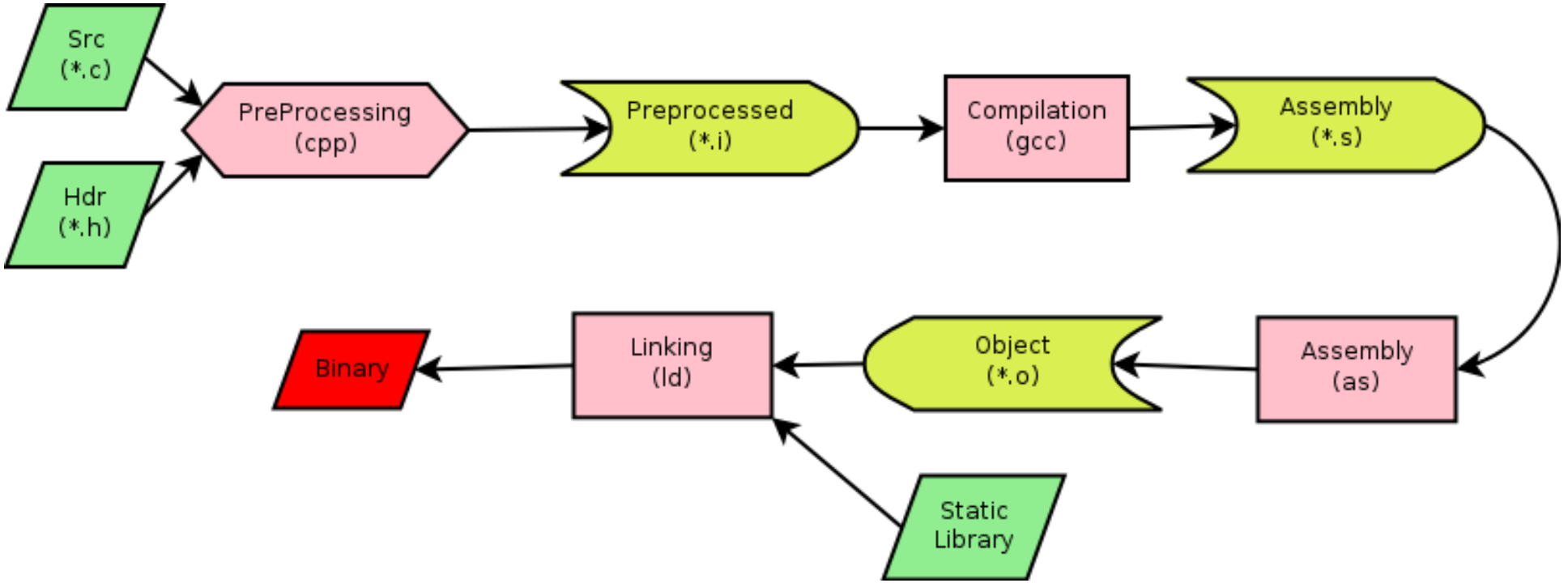
Decompressing Files

- Generally, you receive Linux software in the tarball format (.tgz) or (.gz)
- Decompress file in current directory:
- $ tar –xzvf filename.tar.gz
  - Option –x: --extract
  - Option –z: --gzip
  - Option –v: --verbose
  - Option –f: --file

Build Process

- **configure**
  - Script that checks details about the machine before installation
    - Dependency between packages
  - Creates ‘Makefile’
- **make**
  - Requires ‘Makefile’ to run
  - Compiles all the program code and creates executables in current temporary directory
- **make install**
  - make utility searches for a label named install within the Makefile, and executes only that section of it
  - executables are copied into the final directories (system directories)

GCC Compilation Process



Command-Line Compilation

- item.h
- item.c
  - #include item.h
- shoppingList.h
  - #include item.h
- shoppingList.c
  - #include shoppingList.h
- shop.h
- shop.c
  - #includes shoppingList.h and shop.h
- How to compile?
  - **gcc -Wall shoppingList.c item.c shop.c –o shop**

Expanding the command

- gcc - compiler program
- -Wall - turn all warnings on
- -o - to name the executable as the name given, instead of a.out

What if…

- **We change one of the header or source files?**
  - Rerun command to generate new executable
- **We only made a small change to item.c?**
  - not efficient to recompile shoppinglist.c and shop.c
  - Solution: avoid waste by producing a separate object code file for each source file
    - gcc -Wall –c item.c… (for each source file)
    - gcc item.o shoppingList.o shop.o –o shop (combine)
    - Less work for compiler, saves time but more commands

What if…

- **We change item.h?**
  - Need to recompile every source file that includes it & every source file that includes a header that includes it. Here: item.c, shoppinglist.c and shop.c
  - Difficult to keep track of files when project is large
    - Windows 7 ~40 million lines of code

=> Make

Make

- Utility for managing large software projects
- Compiles files and keeps them up-to-date
- Efficient Compilation (only files that need to be recompiled)

Makefile Example

```
# Makefile - A Basic Example
all : shop  #usually first
shop : item.o shoppingList.o shop.o
        gCC -Wall -o shop item.o shoppingList.o shop.o
item.o : item.cpp item.h
        gCC -Wall -c item.cpp
shoppingList.o : shoppingList.cpp shoppingList.h
        gCC -Wall -c shoppingList.cpp
shop.o : shop.cpp item.h shoppingList.h
        gCC -Wall -c shop.cpp
clean :
        rm -f item.o shoppingList.o shop.o shop
```

Rule

Comments

Targets

Prerequisites

Dependency Line

Commands

Makefile Task

- Create files main.cpp , message.h and message.cpp
- Compile everything together with a single command and execute it
- Now create a simple makefile to compile these files
- Run the executable and observe the output
- Invoke make again and observe
- Change message.cpp, do a make again and observe the output
- Now suppose you want make to re-compile a file without editing the file, which command will you use? Do it for main.cpp