

# Cscope

---

Concurrent Programming

# Introduction

---

- What is the Cscope?
- Installing Cscope
- How to use?
- Practice
- Tip

# What is the Cscope?

- Developer's tool for browsing source code
- Allows searching code for
  - All references to a symbol
  - Global definition
  - Functions called by a function
  - Functions calling a function
  - Text string
  - Regular expression pattern
  - A file
  - Files including a file

# Installing Cscope

```
$ sudo apt-get install cscope
```

- create mkcscope.sh file

```
#!/bin/sh
rm -rf cscope.files
find . \( -name '*.c' -o -name '*.cpp' -o -name '*.cc' -o -name '*.ic' -o -name
 '*.h' -o -name '*.hpp' -o -name '*.s' -o -name '*.S' \) -print > cscope.files
cscope -b -q -k
```

- You can just download this file from Piazza page

# Installing Cscope

- Set permission & path of mkcscope.sh

```
$ chmod 755 mkcscope.sh  
$ sudo mv mkcscope.sh /usr/bin/
```

- Hot key mapping for vim
  - Download 'append\_this\_to\_your\_vimrc' from Piazza page, and append it to your ~/.vimrc

# How to use?

- Generate Cscope database files

```
$ cd project4/mariadb/server/storage/innobase  
$ mkcscope.sh
```

- In the Vim command mode, you can execute the Cscope commands

```
:cs find c|d|e|f|g|i|s|t name
```

# How to use?

- You can use hotkey to navigate a symbol which is under your cursor
  - Because you already modified ~/.vimrc to map hotkey on these Cscope actions
- Hotkey: Ctrl-[, [s|g|c|t|e|f|i|d]
  - 's' symbol: find all references to the token under cursor
  - 'g' global: find global definition(s) of the token under cursor
  - 'c' calls: find all calls to the function name under cursor
  - 't' text: find all instances of the text under cursor
  - 'e' egrep: egrep search for the word under cursor
  - 'f' file: open the filename under cursor
  - 'i' includes: find files that include the filename under cursor
  - 'd' called: find functions that function under cursor calls

# Practice

- Practice with mariadb (on innodb storage engine)
  1. Find “innobase\_init” function definition
  2. Find “trx\_t” structure definition
  3. Find all functions calling “trx\_commit” function
  4. Find a function assigning some value into “log\_sys” variable
  5. Set cursor on a function founded in practice 4.  
Using hotkey, Find a code line that is calling this function (Ctrl-\, c)



# Tip

---

- If you add new symbols, you should generate new Cscope databases(running mkcscope.sh again) to navigate that symbols
- You need to open source files from a directory that contains Cscope database files. Otherwise Cscope cannot recognize symbols you want to navigate.

# Thank You

---