

DEBUG 裝屌指南

那些看起來高級的LLDB手勢

LLDB與我（的前兩年）

```
(lldb) po view.frame
↳ (0.0, 0.0, 375.0, 667.0)
   ▽ origin : (0.0, 0.0)
     - x : 0.0
     - y : 0.0
   ▽ size : (375.0, 667.0)
     - width : 375.0
     - height : 667.0
```

```
print("banner's frame : \$(bannerView.frame)")
```

```
func test() {  
  
    if true { // someCondition1 && someCondition2  
        // do something  
  
    } else {  
        // do others  
  
    }  
}
```

```
func layoutMySubviews() {  
  
    let baseRect| = CGRect(x: 0, y: 0, width: 375, height:  
        100) //calculateBaseRect()  
    // do layout with base rect ...  
}  
  
func calculateBaseRect() → CGRect {  
  
    let rect = CGRect.zero  
  
    // ... do calculation based on screen size  
  
    return rect  
}
```

Compiling was like...

THE #1 PROGRAMMER EXCUSE
FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S COMPILING."



換到Swift 後...



Print Commands

```
print("banner's frame : \$(bannerView.frame)")
```

- p , print, call : expression —

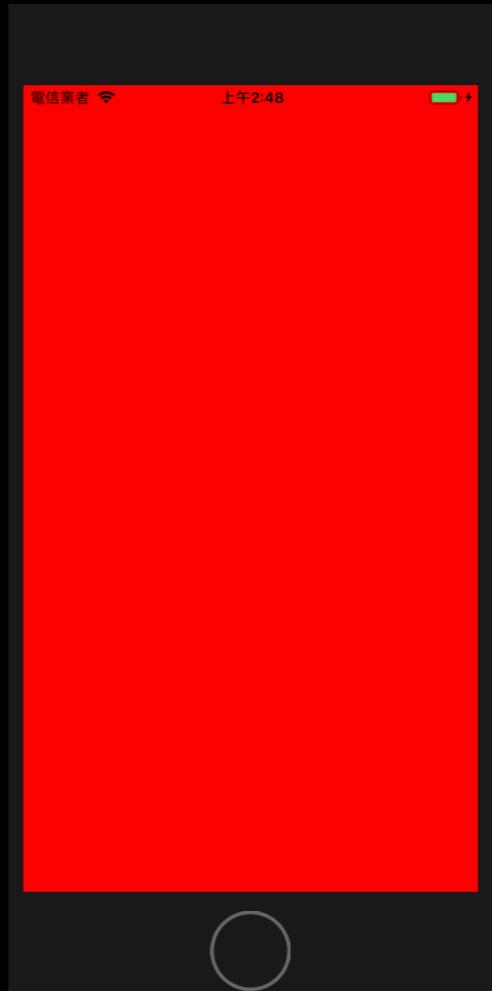
```
(lldb) print count  
(Int) $R0 = 10
```

- po: expression -O —

```
(lldb) po self  
<DispatchInProtocol.ViewController: 0x7fd404f05e50>  
  
(lldb) expr -l objc++ -O -- @[@"a", @"b"]  
<__NSArrayI 0x60400022d940>(  
    a,  
    b  
)
```

Not Only ‘Print Commands’

```
(lldb) print self.view.backgroundColor = UIColor.red  
(lldb) print CATransaction.flush()
```



縮寫到唯一配對即可

- 以下兩者等價

```
(lldb) breakpoint set -n main  
Breakpoint 4: 44 locations.
```

```
(lldb) br s -n main  
Breakpoint 5: 44 locations.
```

可宣告變數

```
(lldb) eobjc int $a = 10  
(lldb) po $a  
10
```

```
(lldb) p $a * 19  
(int) $1 = 190
```

\$開頭之變數可存在於 lldb 的空間

可宣告變數

```
(lldb) expr -- NSArray *$array = @[ @"Saturday", @"Sunday",
@@"Monday" ]
```

```
(lldb) expr [$array count]
(NSUInteger) $0 = 3
```

```
(lldb) p [[$array objectAtIndex:2] characterAtIndex:0]
error: no known method '-characterAtIndex:'; cast the message send
to the method's return type
```

需要指定型別

```
(lldb) p (char)[[$array objectAtIndex:2] characterAtIndex:0]
(char) $1 = 'M'
(lldb) p/x (char)[[$array objectAtIndex:2] characterAtIndex:0]
(char) $2 = 0x4d
```

需要指定型別

ObjC

```
(lldb) p KxMenuView *$menu = (KxMenuView *)0x7f8f7bc11ef0
(lldb) po $menu
<KxMenuView: 0x7f8f7bc11ef0; frame = (0 0; 0 0); alpha = 0; layer = <CALayer: 0x60400042f400>>
```

Swift

```
(lldb) expression -l swift -- let $menu2 = unsafeBitCast(0x7f8f7bc11ef0, to: KxMenuView.self)
```

流程控制

423

```
const CGFloat itemWidth = ((!menuItem.enabled && !  
    menuItem.image) ? titleSize.width : maxImageWidth +  
    titleSize.width) + kMarginX * 4;
```

120 - (BOOL) enabled

121 {

122 ➤ return _target != nil && _action != NULL;

123 }

流程控制

(lldb) thread return YES

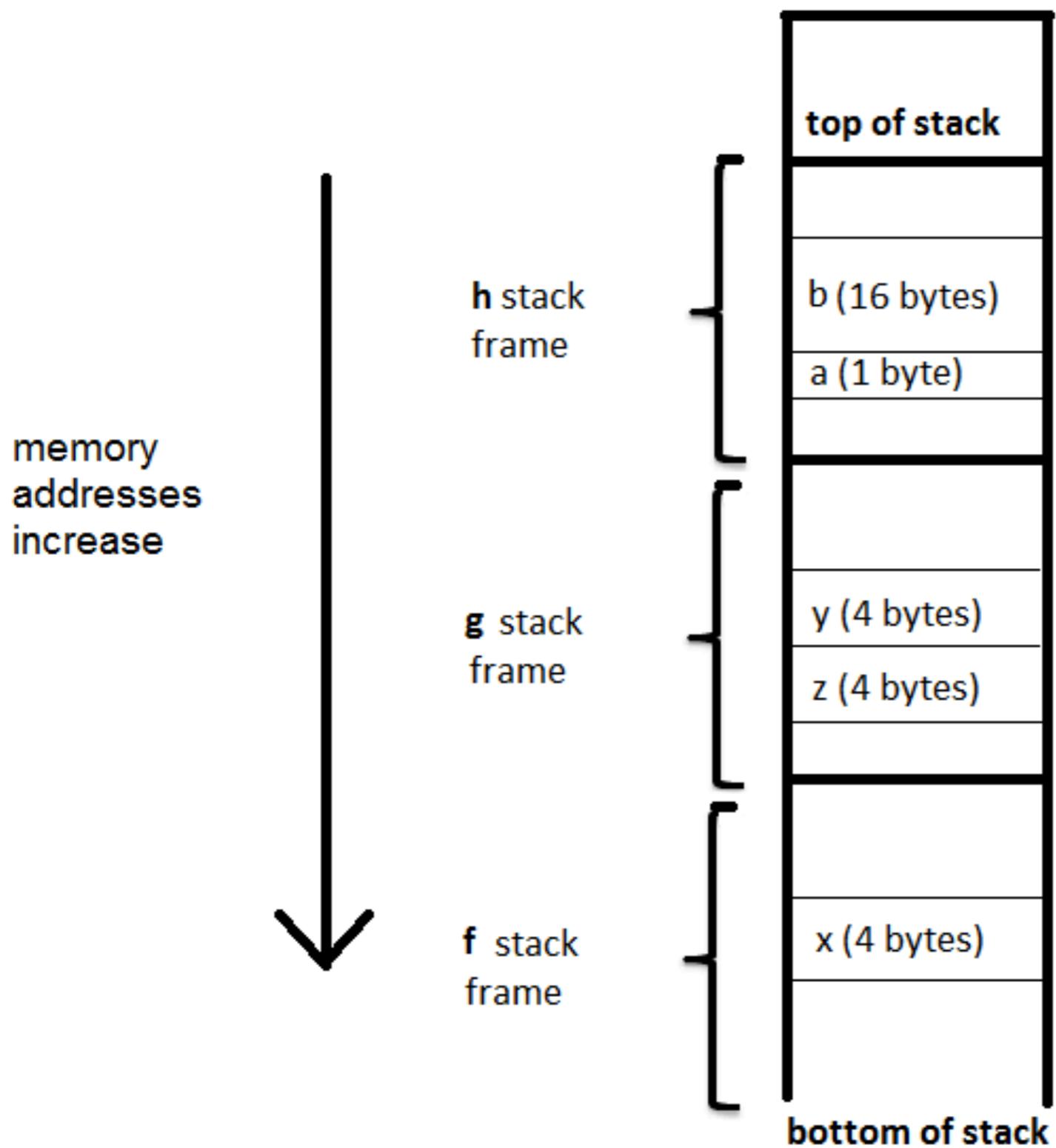
thread backtrace

(lldb) bt

```
* thread #1, queue = 'com.apple.main-thread', stop reason = step over
 * frame #0: 0x0000000104c5a24a KxMenuExample`-[KxMenuView showMenuInView:fromRect:menuItems:]
(self=0x00007f8787416020, _cmd="showMenuInView:fromRect:menuItems:", view=0x00007f8787404ae0, rect=(origin = (x =
5, y = 259), size = (width = 100, height = 50)), menuItems=@"6 elements") at KxMenu.m:320
frame #1: 0x0000000104c5e642 KxMenuExample`-[KxMenu showMenuInView:fromRect:menuItems:]
(self=0x00006000004236e0, _cmd="showMenuInView:fromRect:menuItems:", view=0x00007f8787404ae0, rect=(origin = (x =
5, y = 259), size = (width = 100, height = 50)), menuItems=@"6 elements") at KxMenu.m:829
frame #2: 0x0000000104c5e847 KxMenuExample`+[KxMenu showMenuInView:fromRect:menuItems:](self=KxMenu,
_cmd="showMenuInView:fromRect:menuItems:", view=0x00007f8787404ae0, rect=(origin = (x = 5, y = 259), size = (width
= 100, height = 50)), menuItems=@"6 elements") at KxMenu.m:856
frame #3: 0x0000000104c5823e KxMenuExample`-[ViewController showMenu:](self=0x00007f878760af70,
_cmd="showMenu:", sender=0x00007f878760daa0) at ViewController.m:152
frame #4: 0x0000000104ffe972 UIKit`-[UIApplication sendAction:to:from:
frame #5: 0x000000010517dc3c UIKit`-[UIControl sendAction:to:forEvent:
frame #6: 0x000000010517df59 UIKit`-[UIControl _sendActionsForEvents:
frame #7: 0x000000010517ce86 UIKit`-[UIControl touchesEnded:withEvent:
frame #8: 0x0000000105074807 UIKit`-[UIWindow _sendTouchesForEvent:] +
frame #9: 0x0000000105075f2a UIKit`-[UIWindow sendEvent:] + 4124
frame #10: 0x0000000105019365 UIKit`-[UIApplication sendEvent:] + 352
frame #11: 0x0000000105965a1d UIKit`__dispatchPreprocessedEventFromEve
frame #12: 0x0000000105968672 UIKit`handleEventQueueInternal + 5957
```

▼ Thread 1 Queue: com.apple.main-thread (serial)

- 0 -[KxMenuView showMenuInView:...
- 1 -[KxMenu showMenuInView:from...
- 2 +[KxMenu showMenuInView:from...
- 3 -[ViewController showMenu:]
- 4 -[UIApplication sendAction:to:fro...
- 19 UIApplicationMain
- 20 main
- 21 start
- 22 start



State of stack while **h** is running

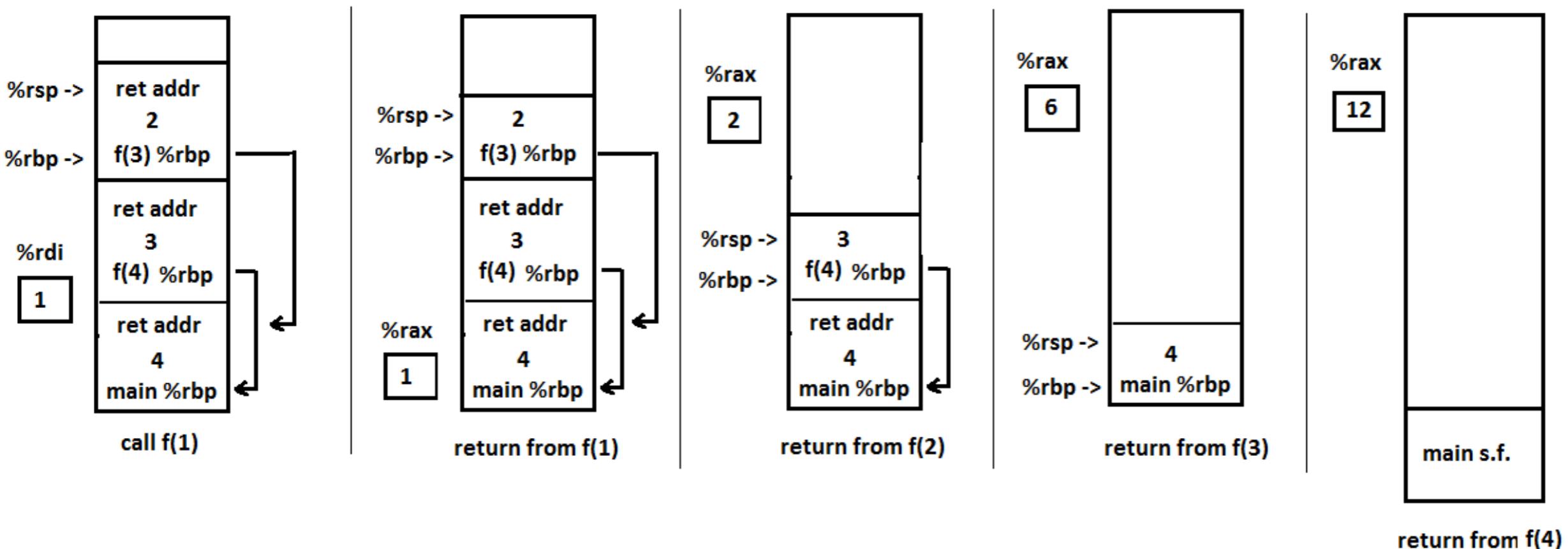
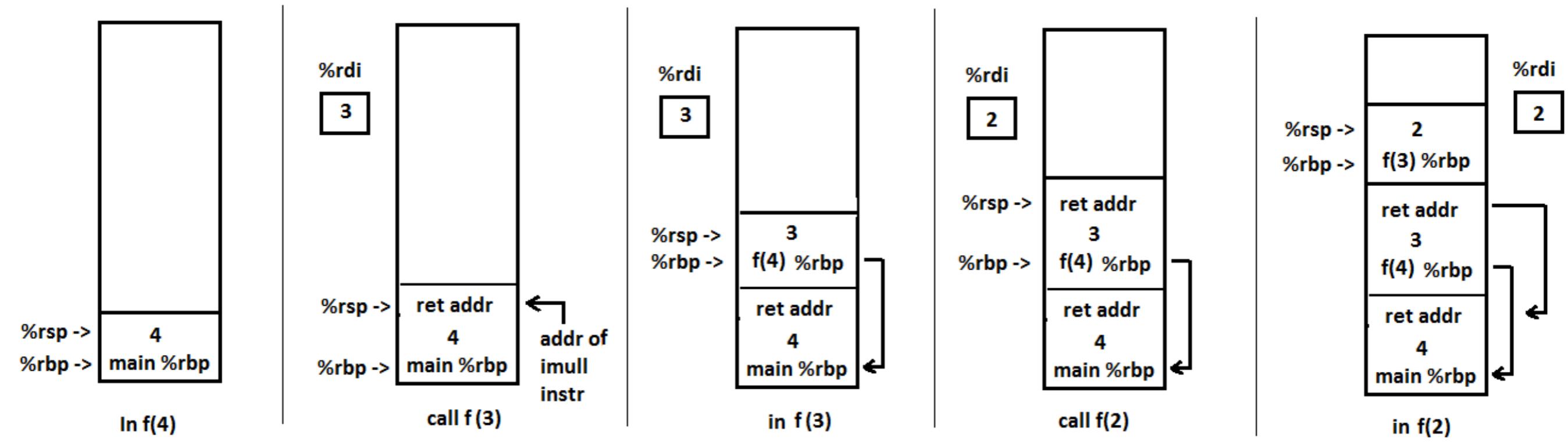
```

void f(...) {
    int x = 5;
    g(x);
}

void g(int x) {
    int y = 10;
    int z = 100;
    h();
}

void h() {
    char a;
    char b[16];
}

```



Frame

(lldb) frame info

```
frame #0: 0x0000000104c5a24a KxMenuExample`-[KxMenuView showMenuInView:fromRect:menuItems:](self=0x00007f8787416020, _cmd="showMenuInView:fromRect:menuItems:", view=0x00007f8787404ae0, rect=(origin = (x = 5, y = 259), size = (width = 100, height = 50)), menuItems=@"6 elements") at
```

(lldb) frame variable

```
(KxMenuView *) self = 0x00007f8787416020
(SEL) _cmd = "showMenuInView:fromRect:menuItems:"
(UIView *) view = 0x00007f8787404ae0
(CGRect) rect = (origin = (x = 5, y = 259), size = (width = 100, height = 50))
(__NSArrayI *) menuItems = 0x0000600000273800 @"6 elements"
(KxMenuOverlay *) overlay = nil
(CGRect) toFrame = (origin = (x = 0, y = 5.4901978502818538E-315), size = (width
```

Breakpoints

- -n : func or symbol name

```
(lldb) breakpoint set -n main
```

```
Breakpoint 4: 44 locations.
```

```
(lldb) br s -n main
```

```
Breakpoint 5: 44 locations.
```

- li : list all breakpoints

```
(lldb) br list
Current breakpoints:
1: file = '/Users/pofat/Code/kxmenu/Source/KxMenu.m', line = 460, exact_match = 0, locations = 1, resolved = 1, hit count = 0
  1.1: where = KxMenuExample`-[KxMenuView mkContentView] + 3626 at KxMenu.m:460, address = 0x0000000104c5bd0a, resolved, hit count = 0
2: file = '/Users/pofat/Code/kxmenu/Source/KxMenu.m', line = 122, exact_match = 0, locations = 1, resolved = 1, hit count = 1
  2.1: where = KxMenuExample`-[KxMenuItem enabled] + 20 at KxMenu.m:122, address = 0x0000000104c58ca4, resolved, hit count = 1
3: file = '/Users/pofat/Code/kxmenu/Source/KxMenu.m', line = 423, exact_match = 0, locations = 1, resolved = 1, hit count = 0
  3.1: where = KxMenuExample`-[KxMenuView mkContentView] + 1869 at KxMenu.m:423, address = 0x0000000104c5b62d, resolved, hit count = 0
```

help & apropos

(lldb) **apropos breakpoint**

The following commands may relate to 'breakpoint':

- | | |
|------------------------|---|
| _regexp-break | -- Set a breakpoint using one of several shorthand formats. |
| _regexp-tbreak | -- Set a one-shot breakpoint using one of several shorthand formats. |
| breakpoint | -- Commands for operating on breakpoints (see 'help b' for shorthand.) |
| breakpoint clear | -- Delete or disable breakpoints matching the specified source file and line. |
| breakpoint command | -- Commands for adding, removing and listing LLDB commands executed when a breakpoint is hit. |
| breakpoint command add | -- Add LLDB commands to a breakpoint, to be executed whenever the breakpoint is hit. If no breakpoint is specified, adds the commands to the last created |

Who did this?

```
var counter = 0 {  
    didSet {  
        check(value: counter)  
    }  
}  
func check(value: Int) {  
  
    // if value == 99 {  
    //     // Terrible things would happen  
    // }  
    // some operation here  
    if value % 2 == 0 {  
        print("even")  
    } else {  
        print("odd")  
    }  
}
```

}

Print? Think again!

```
(lldb) breakpoint set -F check  
Breakpoint 2: where = ConditionalBreakpoi  
33 at ViewController.swift:25, address = 0  
|
```

```
(lldb) breakpoint modify -c 'value == 99' 2  
|
```

```
(lldb) breakpoint command add 2  
Enter your debugger command(s). Type 'DONE' to end.  
> p value  
> bt  
> continue  
> DONE  
|
```

Print? Think again!

鄉民故事你敢信？

明天要上線，merge後的code 不知道怎麼了
覆寫了某個 customs view 的 _layer



Print? Think again!

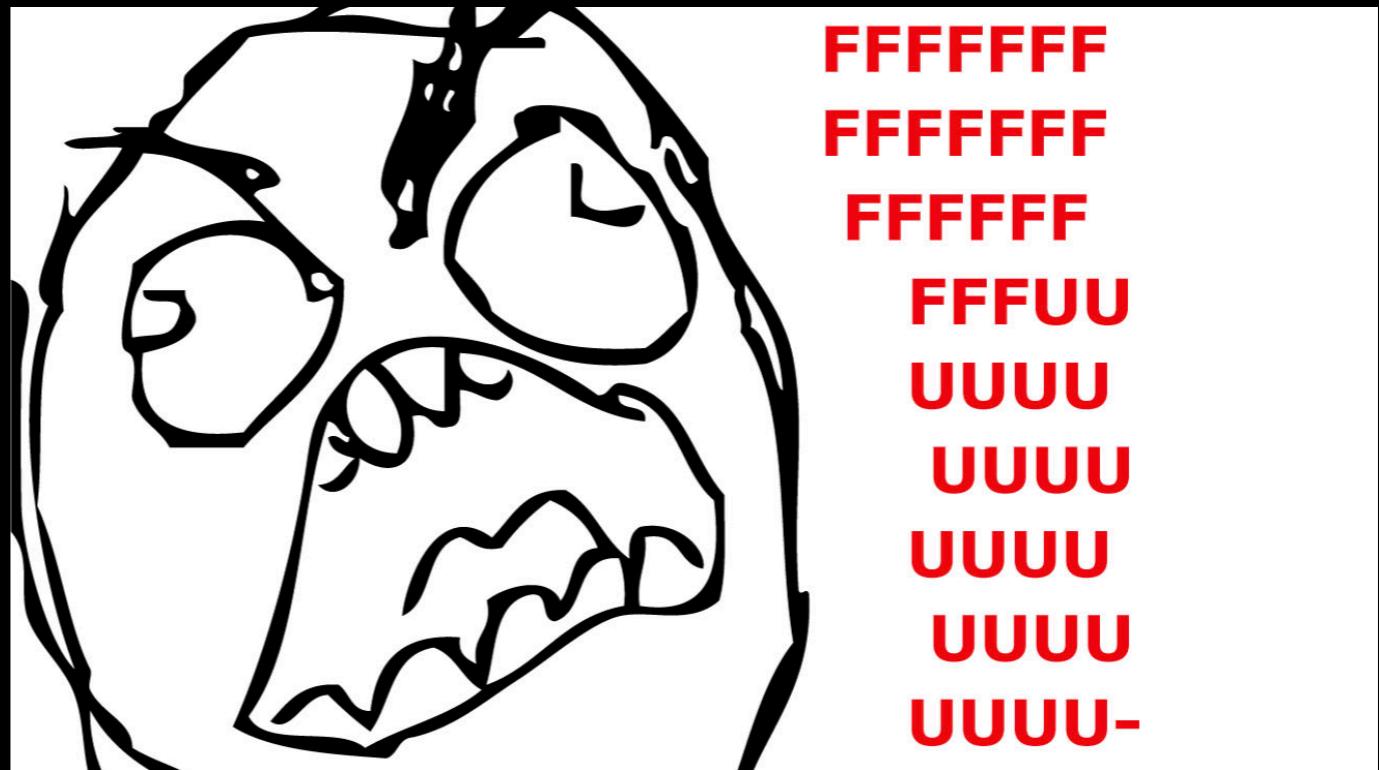
Watchpoint

```
(lldb) p ivar_getOffset((Ivar)class_getInstanceVariable([KxMenuView class], "_layer"))
(ptrdiff_t) $2 = 32
```

```
(lldb) watchpoint set expression -- (int *)$menu + 32
Watchpoint created: Watchpoint 1: addr = 0x7f8f7bc11f70 size = 8 state = enabled type = w
new value: 0x0000000000000000
```

Better way to log

```
print("pofat debug: \\"counter\"\")
```



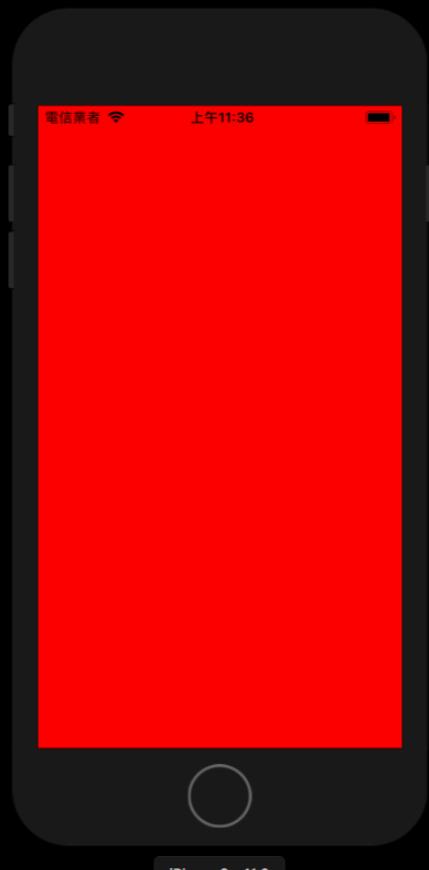
Better way to log maybe

- Breakpoints will minimize the affect to flow
- Need a way to import/export breakpoints



UI 操作 - present VC

```
(lldb) expr UIViewController *$nvc = [[[UIApplication sharedApplication] keyWindow] rootViewController]
(lldb) p UIViewController *$vc = [UIViewController new]
(lldb) p (void)[($vc view) setBackgroundColor: [UIColor redColor]]
(lldb) p (void)[$nvc presentViewController: $vc animated: YES completion: nil]
(lldb) caflush
```



UI 操作 - view hierarchy

```
(lldb) po [[[UIApplication sharedApplication] keyWindow] recursiveDescription]

<UIWindow: 0x7f95a8608740; frame = (0 0; 320 568); gestureRecognizers = <NSArray: 0x600000000000>
| <UIView: 0x7f95a84079d0; frame = (0 0; 320 568); autoresizingMask = W+H; layer = <CALayer: 0x600000000010>
| | <UIButton: 0x7f95a8407dc0; frame = (5 5; 100 50); opaque = NO; layer = <CALayer: 0x600000000020>
| | | <UIButtonLabel: 0x7f95a840b940; frame = (20.5 16; 59 18); text = 'Click me'>
<_UILabelLayer: 0x6040002809b0>>
| | | <UIButton: 0x7f95a860bde0; frame = (5 513; 100 50); opaque = NO; layer = <CALayer: 0x6040002809c0>
| | | | <UIButtonLabel: 0x7f95a840b650; frame = (20.5 16; 59 18); text = 'Click me'>
<_UILabelLayer: 0x604000091c60>>
| | | <UIButton: 0x7f95a860c100; frame = (215 5; 100 50); opaque = NO; layer = <CALayer: 0x604000091c70>
| | | | <UIButtonLabel: 0x7f95a840b360; frame = (20.5 16; 59 18); text = 'Click me'>
<_UILabelLayer: 0x604000091260>>
| | | <UIButton: 0x7f95a860c420; frame = (215 513; 100 50); opaque = NO; layer = <CALayer: 0x604000091270>
```

Viva la Python

```
(lldb) script
Python Interactive Interpreter. To exit, type 'quit()', 'exit()'.

>>> import lldb
>>>
```

.lldbinit

~/command.py

```
1 def caflushCommand(debugger, command, result, internal_dict):  
2     debugger.HandleCommand("e (void)[CATransaction flush]")
```

~/.lldbinit

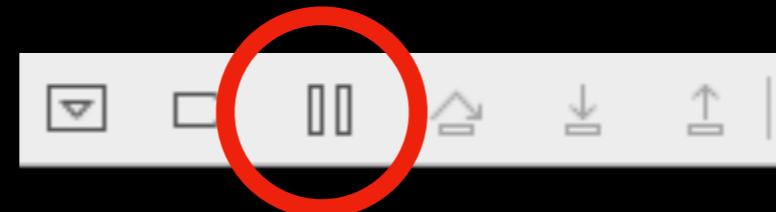
```
6  
7 command script import ~/command.py
```

Chisel

Chisel

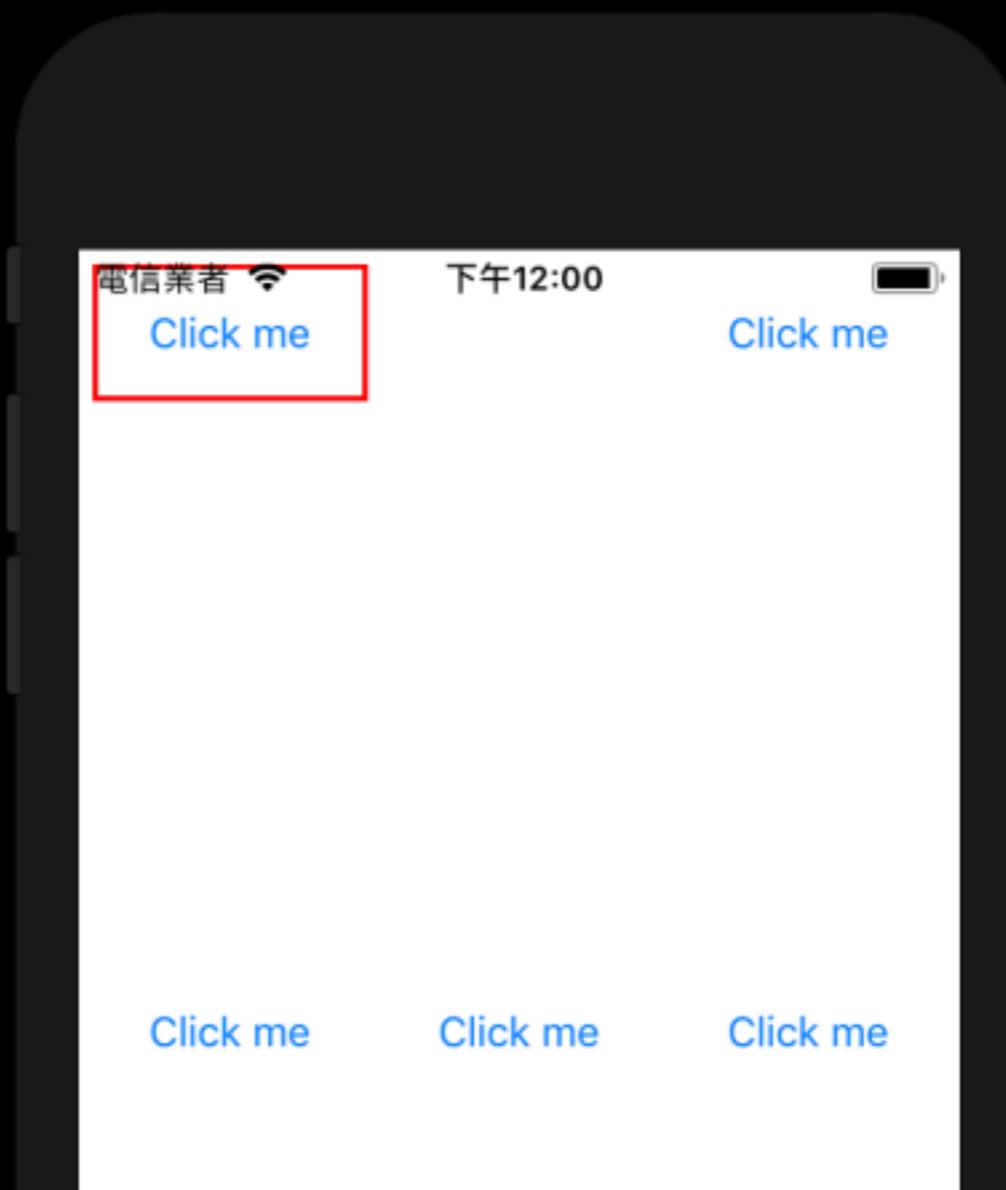
Chisel is a collection of LLDB commands to assist in the debugging of iOS apps.

[[Installation](#) • [Commands](#) • [Custom Commands](#) • [Development Workflow](#) [Contributing](#) • [License](#)]



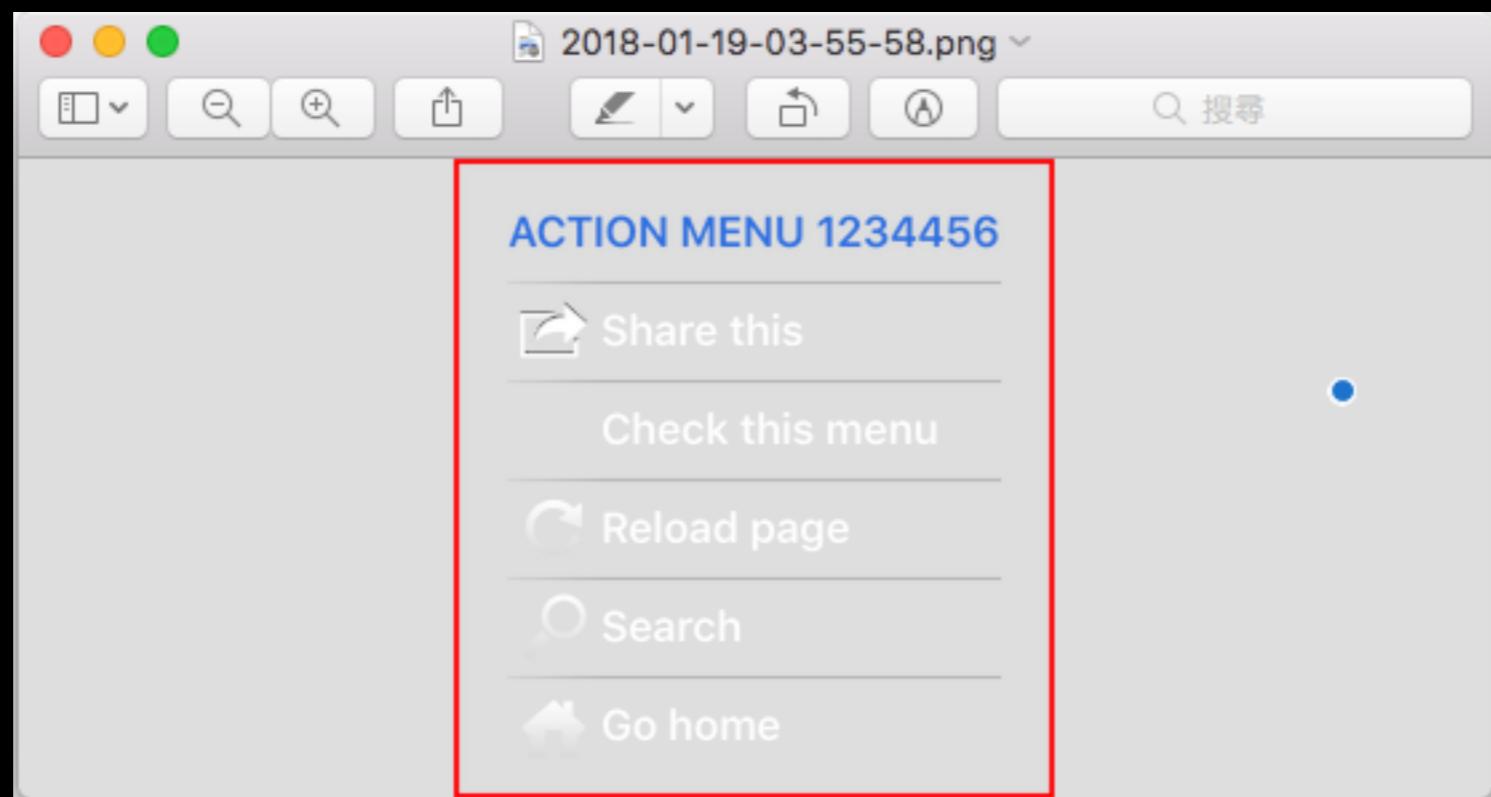
Chisel

- border



Chisel

- visualize



Chisel

- bmessage

```
(lldb) b -[ViewController viewDidLoad]
```

```
Breakpoint 4: no locations (pending).
```

```
WARNING: Unable to resolve breakpoint to any actual locations.
```

```
(lldb) bmessage -[ViewController viewDidLoad]
```

```
Setting a breakpoint at -[UIViewController viewDidLoad] with condition  
(void*)object_getClass((id)$rdi) == 0x0000000109e3f420
```

```
Breakpoint 5: where = UIKit`-[UIViewController viewDidLoad], address = 0x00000
```

Chisel in Swift context

- error: error: use of undeclared identifier 'CGFloat'
- CGFloat is defined by macro in objc. In Swift, it's not.

```
(lldb) e import UIKit
```

Target

```
(lldb) target stop-hook add -o "bt"
```

```
~/.lldbinit
```

```
3 display @import UIKit  
4 target stop-hook add -o "target stop-hook disable"
```

MyPlayground

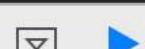
< ! >

```
1 //: Playground - noun: a place where people can play
2
3 import UIKit
4
5 extension Int: Equatable { ! Conformance of 'Int' to protocol 'Equatable' was already stated in...
6     static func ==(lhs: Int, rhs: Int) -> Bool {
7         return true
8     }
9 }
10
11 var a = 0
12
13 if (a == 1) && (a == 2) && (a == 3) {
14     print("hello world")
15 } else {
16     print("no")
17 }
18
```

(3 times)

0

"hello world\n"



hello world

Image lookup

```
(lldb) image lookup -v -F ==
2 matches found in /Users/pofat/Library/Developer/Xcode/DerivedData/FakeExtension-FakeExtension.app/FakeExtension:
    Address: FakeExtension[0x0000000100002830] (FakeExtension.__TEXT.__text +
    Summary: FakeExtension`protocol witness for static Swift.Equatable== inf:
__C.UIApplicationLaunchOptionsKey : Swift.Equatable in UIKit at AppDelegate.swift
    Module: file = "/Users/pofat/Library/Developer/Xcode/DerivedData/FakeExtensi
iphonesimulator/FakeExtension.app/FakeExtension", arch = "x86_64"
    CompileUnit: id = {0x00000000}, file = "/Users/pofat/Desktop/iOSDC/FakeExtens
    Function: id = {0x200000637}, name = "() -> ()", range = [0x0000000105c6e83
    FuncType: id = {0x200000637}, byte-size = 8, compiler_type = "() -> ()"
"
    Blocks: id = {0x200000637}, range = [0x105c6e830-0x105c6e849]
    LineEntry: [0x0000000105c6e830-0x0000000105c6e834]: /Users/pofat/Desktop/iOS
    Symbol: id = {0x00000114}, range = [0x0000000105c6e830-0x0000000105c6e850]
Swift.Bool in conformance __C.UIApplicationLaunchOptionsKey : Swift.Equatable in U
mangled="_T0SC29UIApplicationLaunchOptionsKeyVs9Equatable5UIKitACP2eeoiSbx_xtFZTV
    Address: FakeExtension[0x0000000100002ff0] (FakeExtension.__TEXT.__text +
    Summary: FakeExtension`static (extension in FakeExtension):Swift.Int== inf:
    Module: file = "/Users/pofat/Library/Developer/Xcode/DerivedData/FakeExtensi
iphonesimulator/FakeExtension.app/FakeExtension", arch = "x86_64"
    CompileUnit: id = {0x00000000}, file = "/Users/pofat/Desktop/iOSDC/FakeExtens
```

```
(lldb) image lookup -v -F ==
2 matches found in /Users/pofat/Library/Developer/Xcode/DerivedData/FakeExtension-ehsdvacavwwmezaeccipcklrfpdb/Build/Products/Debug-iphonesimulator/FakeExtension.app/FakeExtension:
    Address: FakeExtension[0x0000000100002830] (FakeExtension.__TEXT.__text + 5440)
    Summary: FakeExtension`protocol witness for static Swift.Equatable== infix(A, A) -> Swift.Bool in conformance
__C.UIApplicationLaunchOptionsKey : Swift.Equatable in UIKit at AppDelegate.swift
    Module: file = "/Users/pofat/Library/Developer/Xcode/DerivedData/FakeExtension-ehsdvacavwwmezaeccipcklrfpdb/Build/Products/Debug-iphonesimulator/FakeExtension.app/FakeExtension", arch = "x86_64"
    CompileUnit: id = {0x00000000}, file = "/Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/AppDelegate.swift", language = "swift"
        Function: id = {0x200000637}, name = "() -> ()", range = [0x0000000105c6e830-0x0000000105c6e849)
        FuncType: id = {0x200000637}, byte-size = 8, compiler_type = "() -> ()"
    "
        Blocks: id = {0x200000637}, range = [0x105c6e830-0x105c6e849)
        LineEntry: [0x0000000105c6e830-0x0000000105c6e834): /Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/AppDelegate.swift
            Symbol: id = {0x00000114}, range = [0x0000000105c6e830-0x0000000105c6e850), name="protocol witness for static Swift.Equatable== infix
Swift.Bool in conformance __C.UIApplicationLaunchOptionsKey : Swift.Equatable in UIKit",
mangled="_T0SC29UIApplicationLaunchOptionsKeyVs9Equatable5UIKitsACP2eeoiSbx_xtFZTW"
        Address: FakeExtension[0x0000000100002ff0] (FakeExtension.__TEXT.__text + 7424)
        Summary: FakeExtension`static (extension in FakeExtension):Swift.Int== infix(Swift.Int, Swift.Int) -> Swift.Bool at MyClass.swift:23
        Module: file = "/Users/pofat/Library/Developer/Xcode/DerivedData/FakeExtension-ehsdvacavwwmezaeccipcklrfpdb/Build/Products/Debug-iphonesimulator/FakeExtension.app/FakeExtension", arch = "x86_64"
        CompileUnit: id = {0x00000000}, file = "/Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/MyClass.swift", language = "swift"
        Function: id = {0x300000255}, name = "(Swift.Int, Swift.Int) -> Swift.Bool", range = [0x0000000105c6eff0-0x0000000105c6f000)
        FuncType: id = {0x300000255}, byte-size = 8, decl = MyClass.swift:23, compiler_type = "(Swift.Int, Swift.Int) -> Swift.Bool"
    "
        Blocks: id = {0x300000255}, range = [0x105c6eff0-0x105c6f000)
        LineEntry: [0x0000000105c6eff0-0x0000000105c6eff4): /Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/MyClass.swift:23
            Symbol: id = {0x00000196}, range = [0x0000000105c6eff0-0x0000000105c6f000), name="static (extension in FakeExtension):Swift.Int== infix
(Swift.Int, Swift.Int) -> Swift.Bool", mangled="_T0Si13FakeExtensionE2eeoiSbSi_SitFZ"
        Variable: id = {0x300000272}, name = "lhs", type = "Swift.Int", location = DW_OP_fbreg(-8), decl = MyClass.swift:23
        Variable: id = {0x300000280}, name = "rhs", type = "Swift.Int", location = DW_OP_fbreg(-16), decl = MyClass.swift:23
        Variable: id = {0x30000028e}, name = "self", type = "@thin Swift.Int.Type", location = , decl = MyClass.swift:23
```

```
ry/Developer/Xcode/DerivedData/FakeExtension-ehsdvacavwwmezaeccipcklrfpdb/Build/Products/Debug-iphonesimu  
0010002830] (FakeExtension.__TEXT.__text + 5440)  
    witness for static Swift.Equatable== infix(A, A) -> Swift.Bool in conformance  
    Swift.Equatable in UIKit at AppDelegate.swift  
/Library/Developer/Xcode/DerivedData/FakeExtension-ehsdvacavwwmezaeccipcklrfpdb/Build/Products/Debug-  
eExtension", arch = "x86_64"  
    file = "/Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/AppDelegate.swift", language = "swift"  
    name = "() -> ()", range = [0x0000000105c6e830-0x0000000105c6e849)  
    byte-size = 8, compiler_type = "() -> ()"  
  
range = [0x105c6e830-0x105c6e849)  
0x0000000105c6e834): /Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/AppDelegate.swift  
range = [0x0000000105c6e830-0x0000000105c6e850), name="protocol witness for static Swift.Equatable== infi  
cationLaunchOptionsKey : Swift.Equatable in UIKit",  
ionsKeyVsEquatable5UITKitsACP2eeoiSbx xtF7TW"  
0010002ff0] (FakeExtension.__TEXT.__text + 7424)  
(extension in FakeExtension):Swift.Int== infix(Swift.Int, Swift.Int) -> Swift.Bool at MyClass.swift:23  
/Library/Developer/Xcode/DerivedData/FakeExtension-ehsdvacavwwmezaeccipcklrfpdb/Build/Products/Debug-  
eExtension", arch = "x86_64"  
    file = "/Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/MyClass.swift", language = "swift"  
    name = "(Swift.Int, Swift.Int) -> Swift.Bool", range = [0x0000000105c6eff0-0x0000000105c6f000)  
    byte-size = 8, decl = MyClass.swift:23, compiler_type = "(Swift.Int, Swift.Int) -> Swift.Bool"  
  
range = [0x105c6eff0-0x105c6f000)  
0x0000000105c6eff4): /Users/pofat/Desktop/iOSDC/FakeExtension/FakeExtension/MyClass.swift:23  
range = [0x0000000105c6eff0-0x0000000105c6f000), name="static (extension in FakeExtension):Swift.Int==  
Bool", mangled="_T0Si13FakeExtensionE2eeoiSbSi_SitFZ"  
name = "lhs", type = "Swift.Int", location = DW_OP_fbreg(-8), decl = MyClass.swift:23  
name = "rhs", type = "Swift.Int", location = DW_OP_fbreg(-16), decl = MyClass.swift:23  
name = "self", type = "@thin Swift.Int.Type", location = , decl = MyClass.swift:23
```

Image lookup

```
(lldb) image lookup -t Logger
```

```
Best match found in /Users/pofat/Library/Developer/Xcode/DerivedData/
Debug-iphonesimulator/ [REDACTED] :
id = {0x0005639f}, name = " [REDACTED].Logger", byte-size = 8, decl = Logger.
  class var sharedInstance: [REDACTED].Logger {
    get {}
}
final let path: Swift.String
var outputStream: <null>
init()
func initOutputStream()
@objc_deinit
func writeLogLine(selectedCellIndex: Swift.Int, selectedCellText: Swift.Str
func writeLogLine(tappedKey: SPiKeyboard.Key)
func writeLogLine(filledString: Swift.String)
func writeLogFileWith(_ string: Swift.String)
func getLogFileContent() -> Swift.String
func clearLogFile()
func getMemoryUsageReport() -> Swift.String
}
```

Image lookup

- 可找私有 API
- 也可在 lldb 裡做 method swizzling

(lldb) image lookup -rn UIApplication

```
Summary: UIKit`-[UIApplication _systemNavigationAction]           Address: UIKit[0x000000000002f1]
Summary: UIKit`-[UIApplication _setSystemNavigationAction:]        Address: UIKit[0x00000000000000]
Summary: UIKit`__44-[UIApplication _setSystemNavigationAction:]_block_invoke   Address: U
,
53)   Summary: UIKit`__44-[UIApplication _setSystemNavigationAction:]_block_invoke.1813      Address:
Summary: UIKit`-[UIApplication _clearSystemNavigationAction]          Address: UIKit[0x00000000000000]
```

Registers

- \$rip : Current instruct pointer
- \$rdi: First argument
- \$rsi, \$rdx, \$rcx, \$R8, \$R9: Second, third, forth, fifth and sixth arguments
- Others will be on stack

Registers

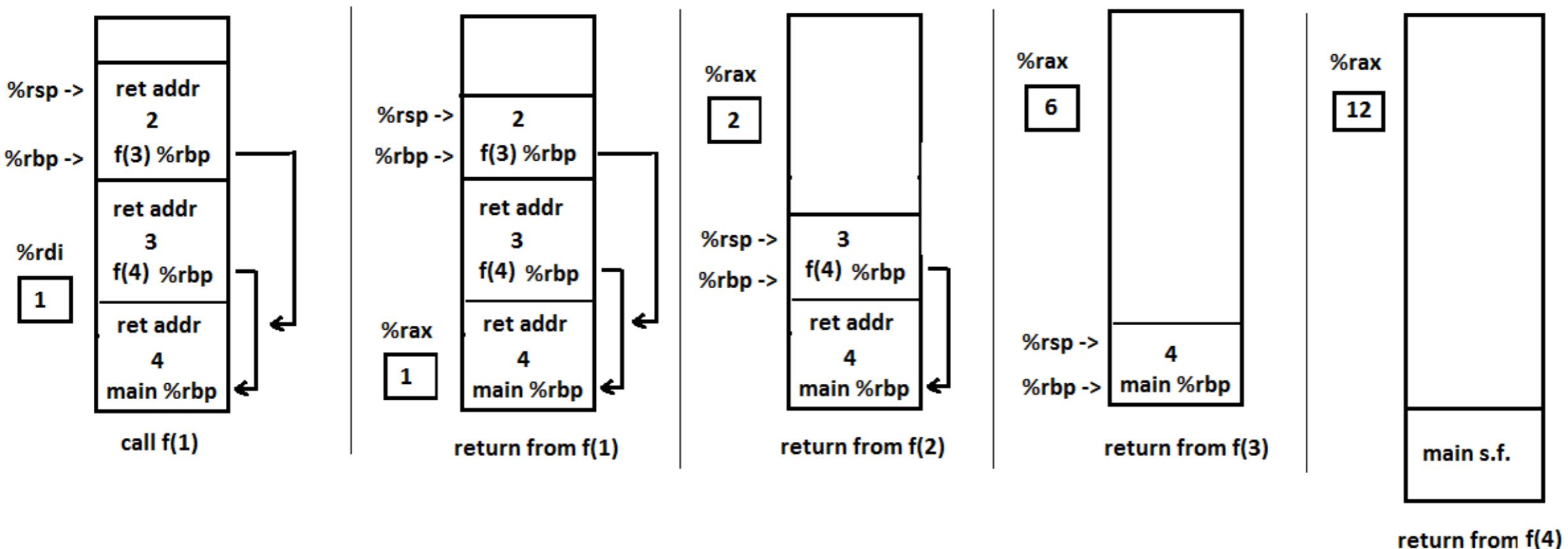
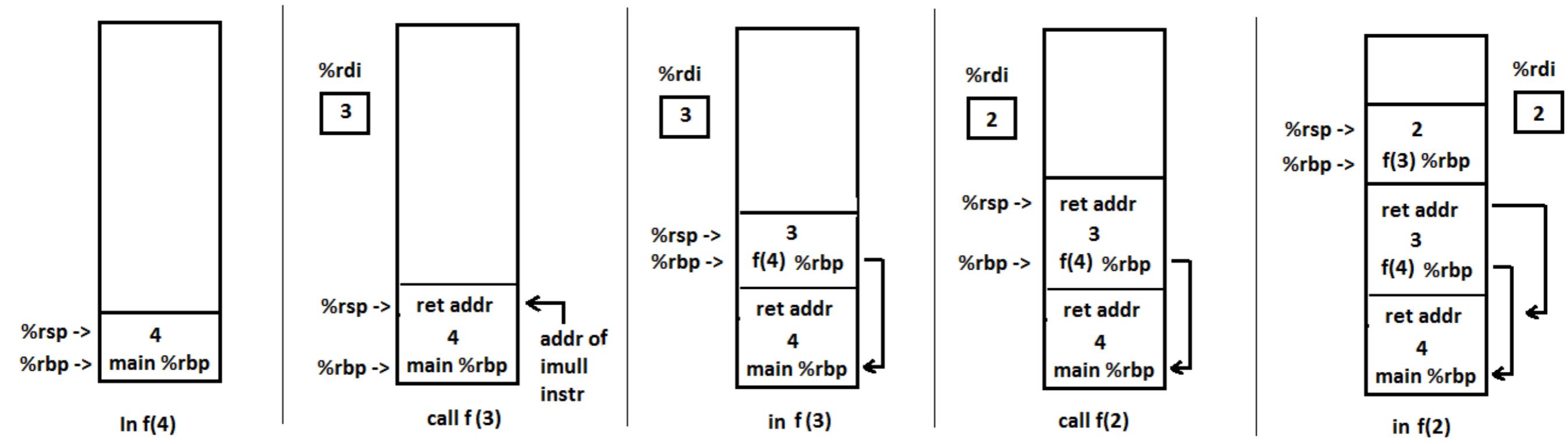
- \$rip : Current instruct pointer
- \$rdi: First argument
- \$rsi, \$rdx, \$rcx, \$R8, \$R9: Second, third, forth, fifth and sixth arguments
- Others will be on stack

Registers

- \$rsp : head of current frame (head is smaller than bottom)
- \$rbp: bottom of current frame
- \$rax: Return register

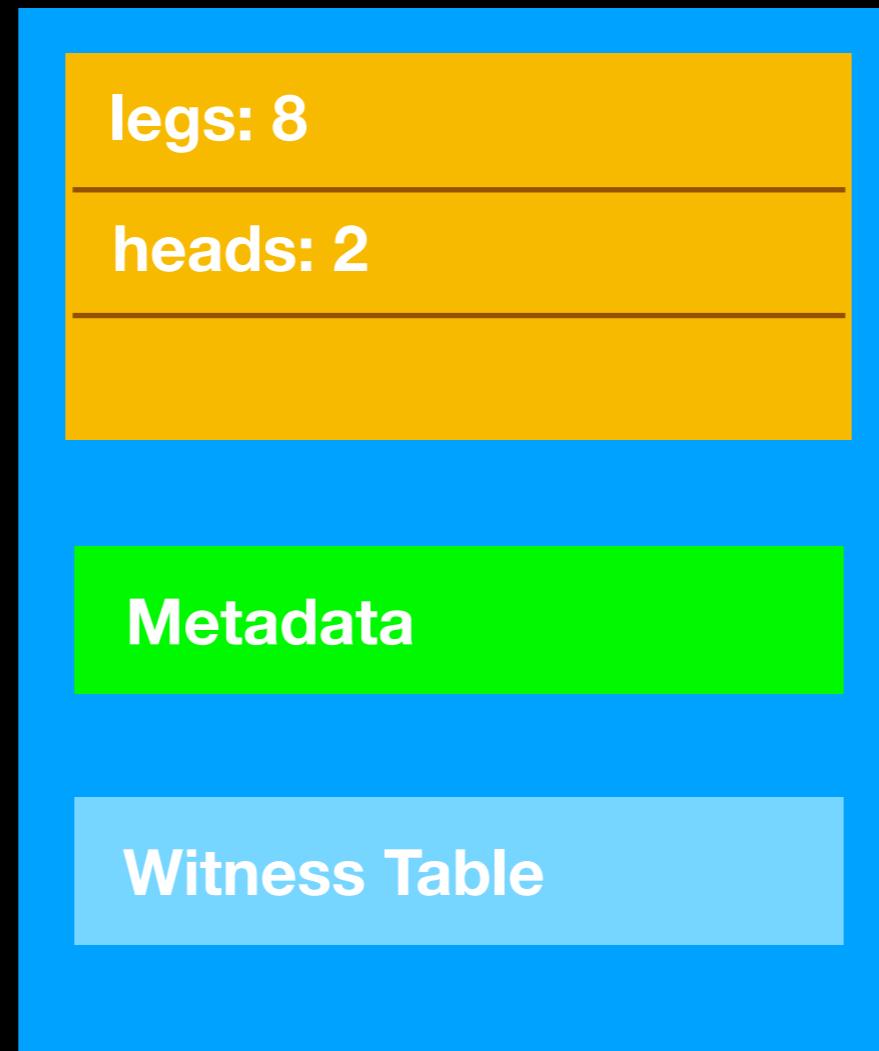
x86-64 Integer Registers: Usage Conventions

%rax	Return value	%r8	Argument #5
%rbx	Callee saved	%r9	Argument #6
%rcx	Argument #4	%r10	Caller saved
%rdx	Argument #3	%r11	Caller Saved
%rsi	Argument #2	%r12	Callee saved
%rdi	Argument #1	%r13	Callee saved
%rsp	Stack pointer	%r14	Callee saved
%rbp	Callee saved	%r15	Callee saved

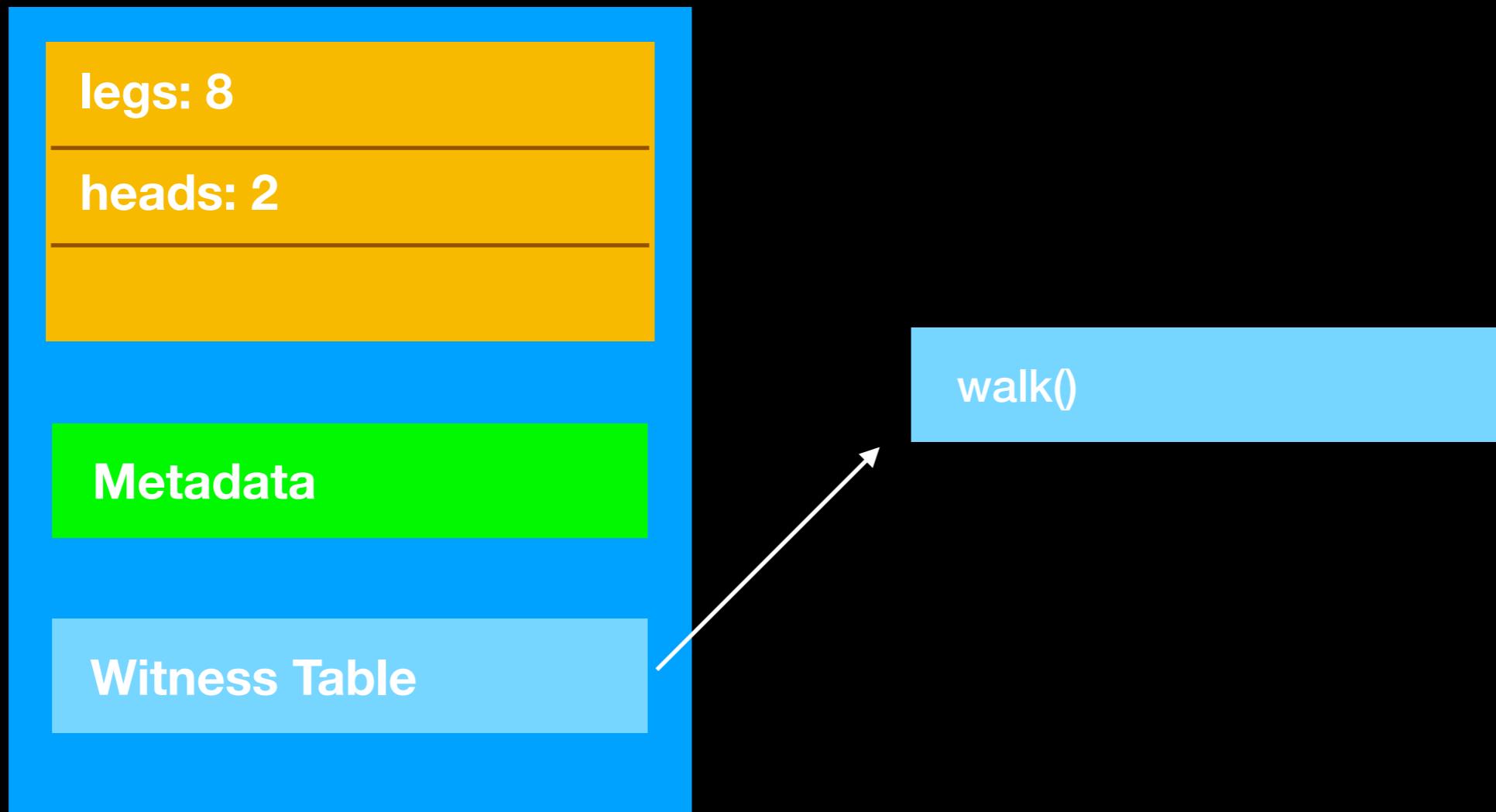


**Witness “Witness
Table”**

Witness “Witness Table”



Witness “Witness Table”



Output all current state

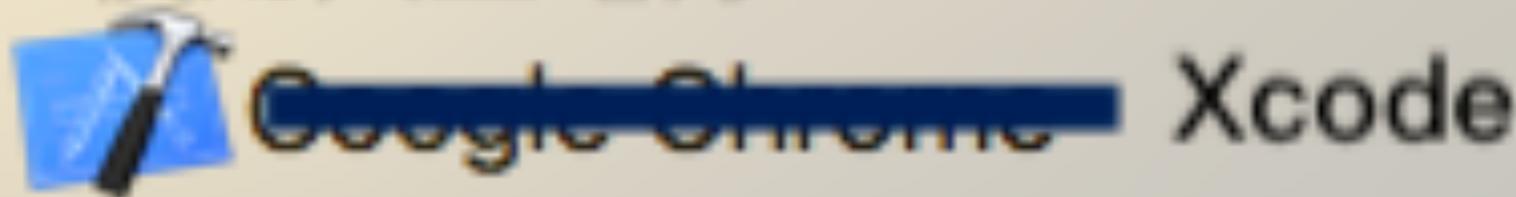
(lldb) bugreport unwind --outfile <output path>

```
1 (lldb) thread backtrace
2 * thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 3.1
3   * frame #0: 0x00000001022278e0 DispatchInProtocol`ViewController.viewDidLoad(self=0x00007fa1d9507450) at ViewController.swift:24
4     frame #1: 0x0000000102227a34 DispatchInProtocol`@objc ViewController.viewDidLoad() at ViewController.swift:0
5     frame #2: 0x000000010360446c UIKit`-[UIViewController loadViewIfRequired] + 1235
6     frame #3: 0x00000001036048b9 UIKit`-[UIViewController view] + 27
7     frame #4: 0x00000001034cf7cf UIKit`-[UIWindow addRootViewControllerViewIfPossible] + 122
8     frame #5: 0x00000001034cfed7 UIKit`-[UIWindow _setHidden:forced:] + 294
9     frame #6: 0x00000001034e2e54 UIKit`-[UIWindow makeKeyAndVisible] + 42
10    frame #7: 0x00000001034558b8 UIKit`-[UIApplication _callInitializationDelegatesForMainScene:transitionContext:] + 4737
11    frame #8: 0x000000010345aaeb UIKit`-[UIApplication _runWithMainScene:transitionContext:completion:] + 1720
```

解放你的豆腐頭

電力來源：電池

使用大量電力



顯示百分比

打開能源節約器偏好設定…

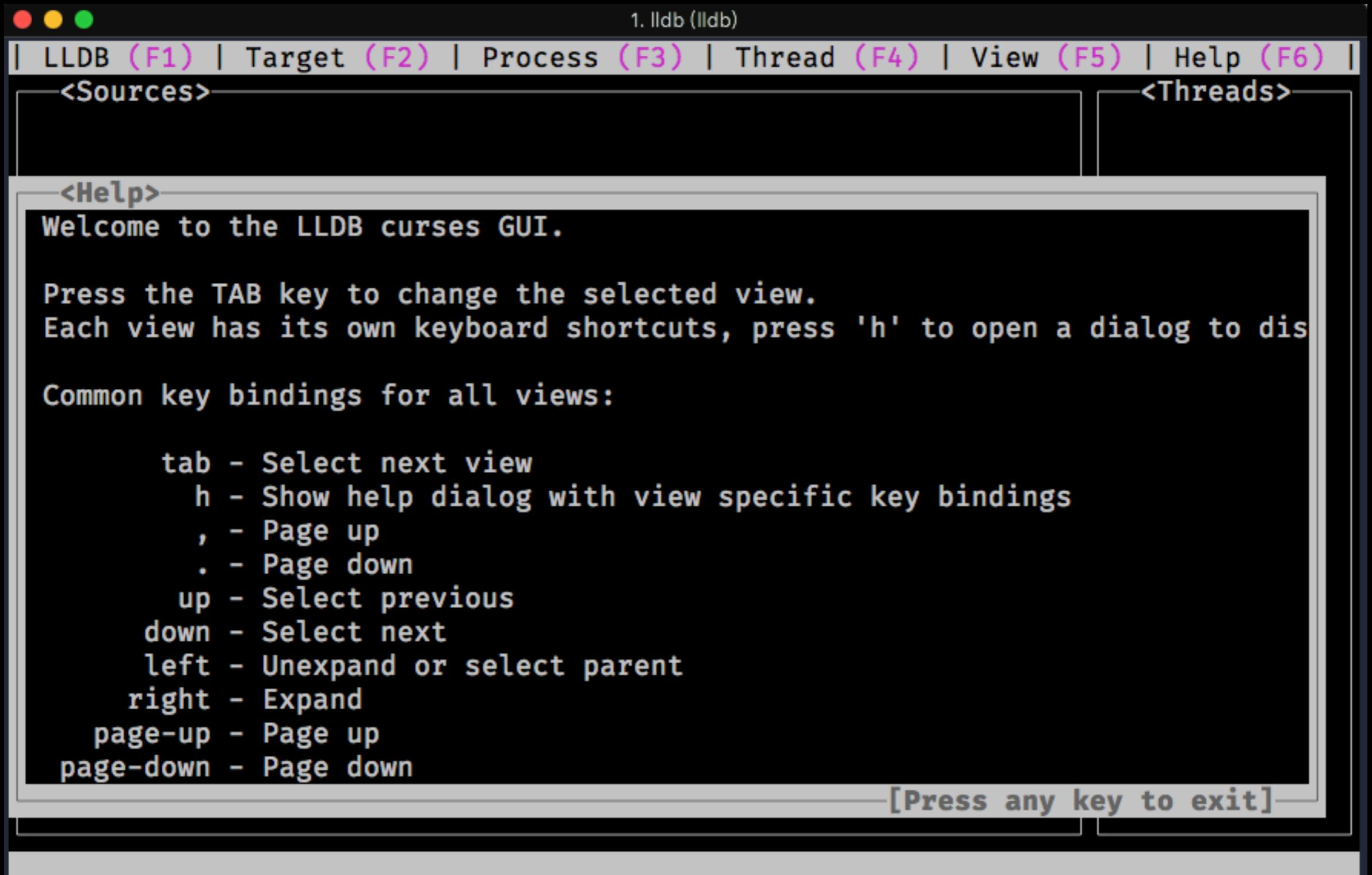
全面command line 啦

```
→ DispatchInProtocol xcrun simctl list | grep "iPhone X"
iPhone X (com.apple.CoreSimulator.SimDeviceType.iPhone-X)
iPhone X (2997C607-8EB3-40C2-9C3C-B444E6F53234) (Shutdown)
iPhone X (5B843BFF-685B-430D-A7F3-FE0754EF11D4) (Shutdown) (unavailable, runtime profi
iPhone X (65B74D42-0D02-4C8D-8FA9-69F12E5898B8) (Shutdown) (unavailable, runtime profi
→ DispatchInProtocol open /Applications/Xcode.app/Contents/Developer/Applications/Simulat
2997C607-8EB3-40C2-9C3C-B444E6F53234
→ DispatchInProtocol lldb -n SpringBoard
```

```
(lldb) image dump sections DispatchInProtocol
Sections for '/Users/pofat/Library/Developer/CoreSimulator/Devices/2A3D4699-ADDE-4EE8-9918-1B59BEED6F71/data/Containers/Bundle
SectID      Type          Load Address                                Perm File Off.  File Size  Flags        Section Name
-----      -----          -----                                -----  -----  -----  -----  -----
0x00000100  container    [0x0000000000000000-0x0000000100000000)* --- 0x00000000 0x00000000 0x00000000 DispatchInProtocol
0x00000200  container    [0x000000010729b000-0x00000001072a1000) r-x 0x00000000 0x00006000 0x00000000 DispatchInProtocol
0x00000001  code         [0x000000010729c9f0-0x000000010729ee00) r-x 0x00019f0 0x0002410 0x80000400 DispatchInProtocol
0x00000002  code         [0x000000010729ee00-0x000000010729eed8) r-x 0x0003e00 0x00000d8 0x80000408 DispatchInProtocol
0x00000003  code         [0x000000010729eed8-0x000000010729f050) r-x 0x0003ed8 0x00000178 0x80000400 DispatchInProtocol
0x00000004  data-cstr   [0x000000010729f050-0x000000010729fc22) r-x 0x0004050 0x00000bd2 0x00000002 DispatchInProtocol
0x00000005  regular     [0x000000010729fc30-0x000000010729fded) r-x 0x0004c30 0x000001bd 0x00000000 DispatchInProtocol
0x00000006  regular     [0x000000010729fdf0-0x000000010729fe90) r-x 0x0004df0 0x000000a0 0x00000000 DispatchInProtocol
0x00000007  regular     [0x000000010729fe90-0x00000001072a005c) r-x 0x0004e90 0x000001cc 0x00000000 DispatchInProtocol
0x00000008  regular     [0x00000001072a005c-0x00000001072a0074) r-x 0x0000505c 0x00000018 0x00000000 DispatchInProtocol
0x00000009  regular     [0x00000001072a0078-0x00000001072a00e8) r-x 0x00005078 0x00000070 0x00000000 DispatchInProtocol
```



```
(lldb) br set -n "-[UIViewController viewDidLoad]" -c "*(uintptr_t*)$rsp <= 0x00000001072af000 && *(uintptr_t*)$rsp >= 0x000000010729b000
Breakpoint 6: where = UIKit`-[UIViewController viewDidLoad], address = 0x000000010839edf6
```



Which one is better?

```
func draw(_ shape: Drawable) {  
    shape.draw()  
}  
  
func genericDraw<T: Drawable>(_ shape: T) {  
    shape.draw()  
}
```

Reference

- Dancing in the Debugger - A waltz with LLDB
- 泊學 (很精實的中國教學網站)
- WWDC 2014 Session 410 : Advanced Debugging in LLDB

感謝各位