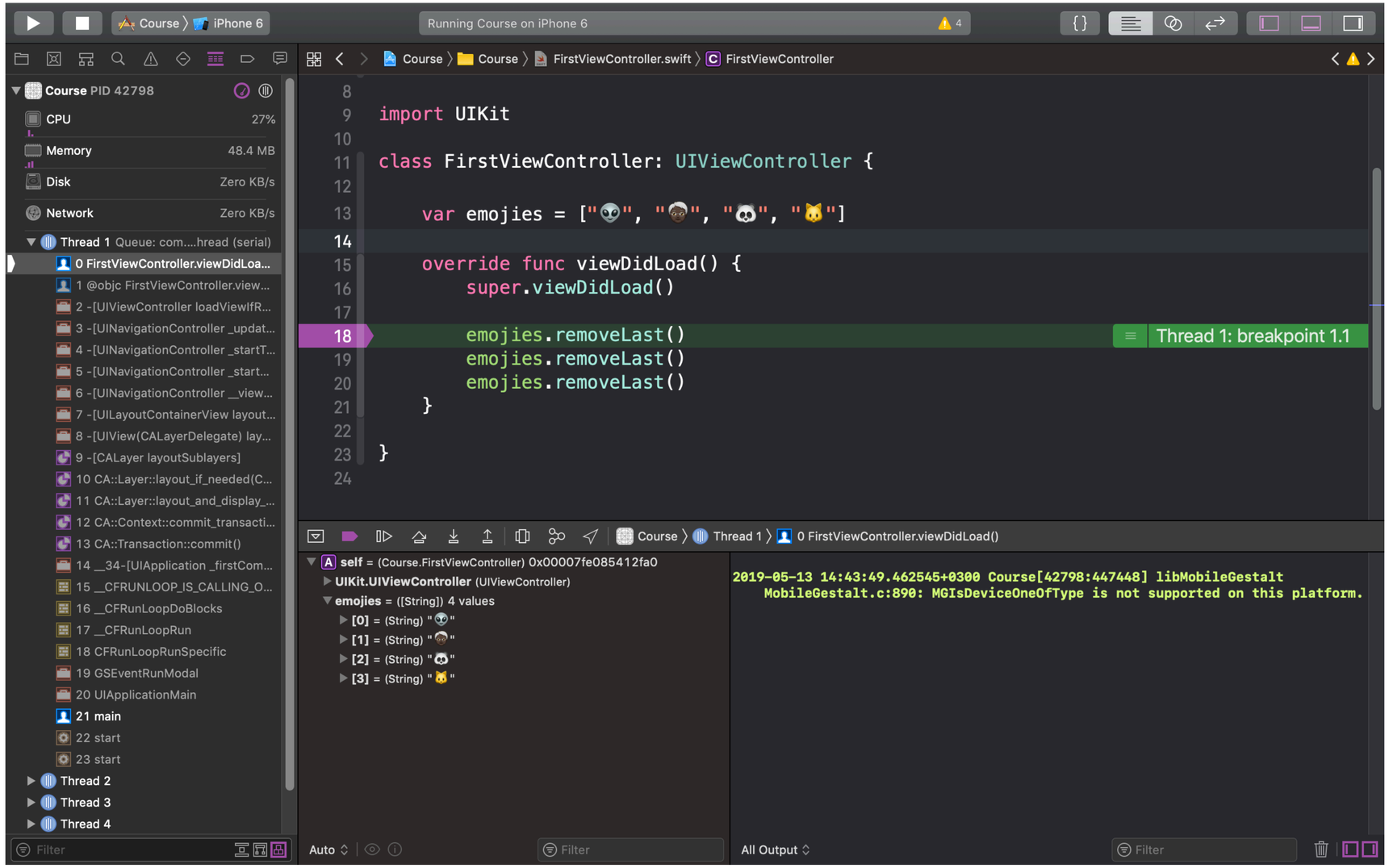
Debugging, Profiling

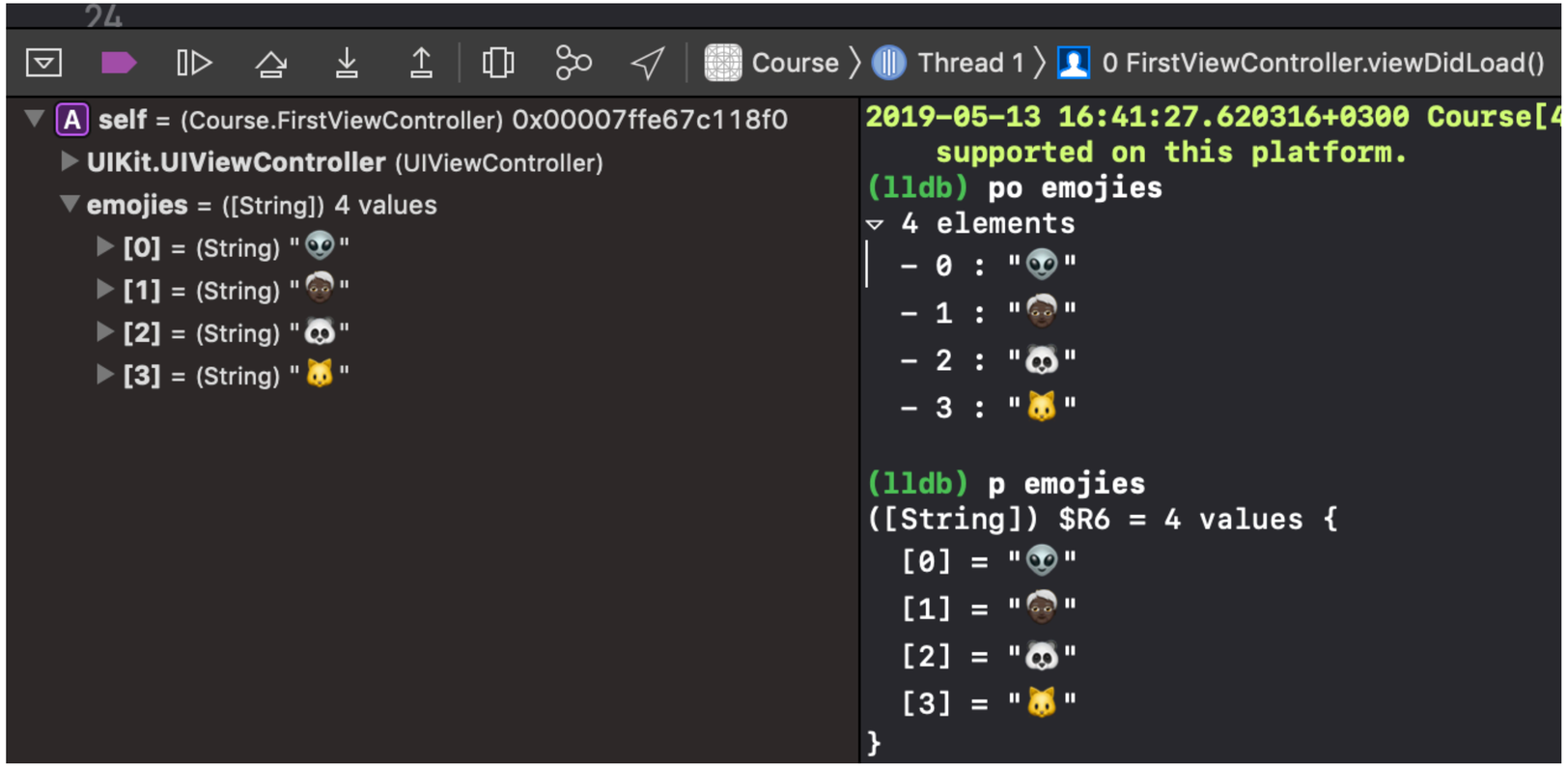
Debugging


Course 〉 iPhone 6
Running Course on iPhone 6
Course 〉 Course 〉 FirstViewController.swift 〉 FirstViewController
Course PID 42798
CPU 27%
Memory 48.4 MB
Disk Zero KB/s
Network Zero KB/s
Thread 1 Queue: com....hread (serial)
0 FirstViewController.viewDidLoa...
1 @objc FirstViewController.view...
2 -[UIViewController loadViewIfR...
3 -[UINavigationController _updat...
4 -[UINavigationController _startT...
5 -[UINavigationController _start...
6 -[UINavigationController __view...
7 -[UILayoutContainerView layout...
8 -[UIView(CALayerDelegate) lay...
9 -[CALayer layoutSublayers]
10 CA::Layer::layout_if_needed(C...
11 CA::Layer::layout_and_display_...
12 CA::Context::commit_transacti...
13 CA::Transaction::commit()
14 __34-[UIApplication _firstCom...
15 __CFRUNLOOP_IS_CALLING_O...
16 __CFRunLoopDoBlocks
17 __CFRunLoopRun
18 CFRunLoopRunSpecific
19 GSEventRunModal
20 UIApplicationMain
21 main
22 start
23 start
Thread 2
Thread 3
Thread 4
Filter
8
9 import UIKit
10
11 class FirstViewController: UIViewController {
12
13 var emojies = ["👽", "👹", "💀", "🐱"]
14
15 override func viewDidLoad() {
16 super.viewDidLoad()
17
18 emojies.removeLast()
Thread 1: breakpoint 1.1
19 emojies.removeLast()
20 emojies.removeLast()
21 }
22
23 }
24
Course 〉 Thread 1 〉 0 FirstViewController.viewDidLoad()
self = (Course.FirstViewController) 0x00007fe085412fa0
UIKit.UIViewController (UIViewController)
emojies = ([String]) 4 values
[0] = (String) "👽"
[1] = (String) "👹"
[2] = (String) "💀"
[3] = (String) "🐱"
2019-05-13 14:43:49.462545+0300 Course[42798:447448] libMobileGestalt
MobileGestalt.c:890: MGIsDeviceOneOfType is not supported on this platform.
Auto
Filter
All Output
Filter

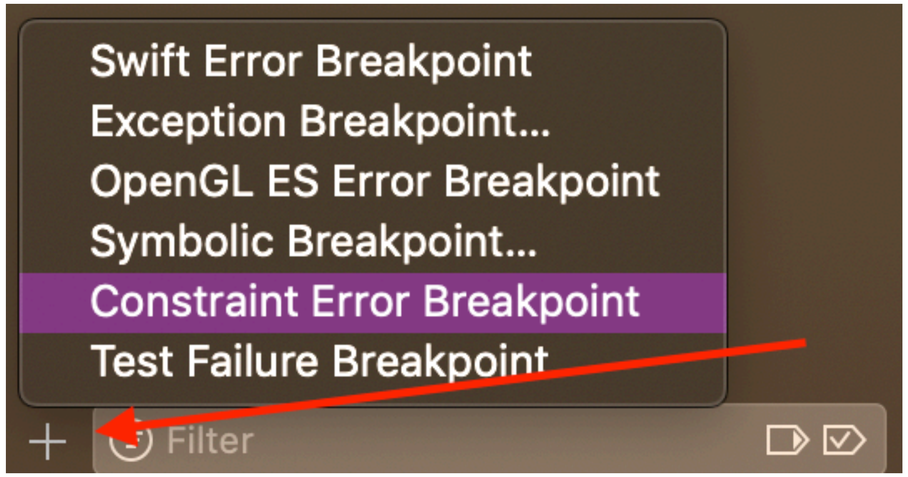
Console debugging

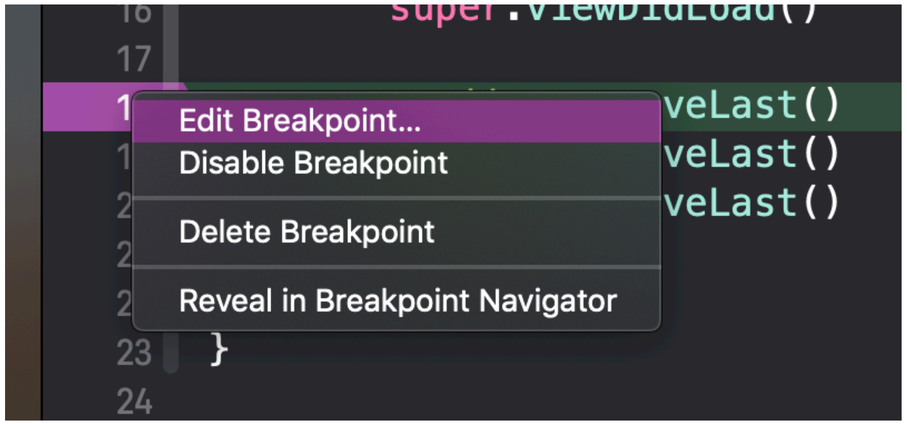
There are two common commands to show object's description in console:

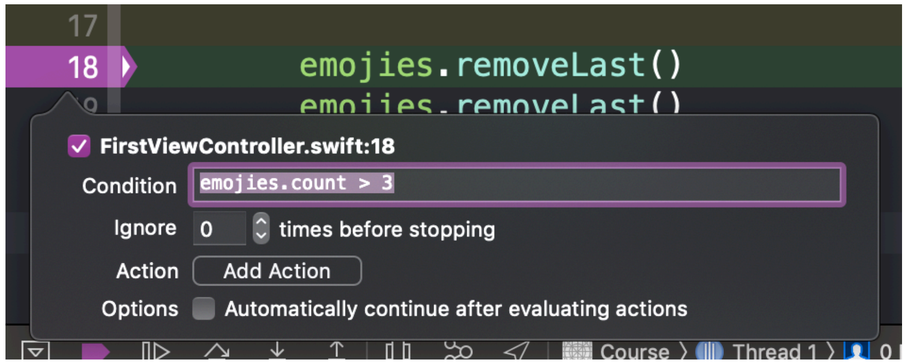
- **p**
- **po**



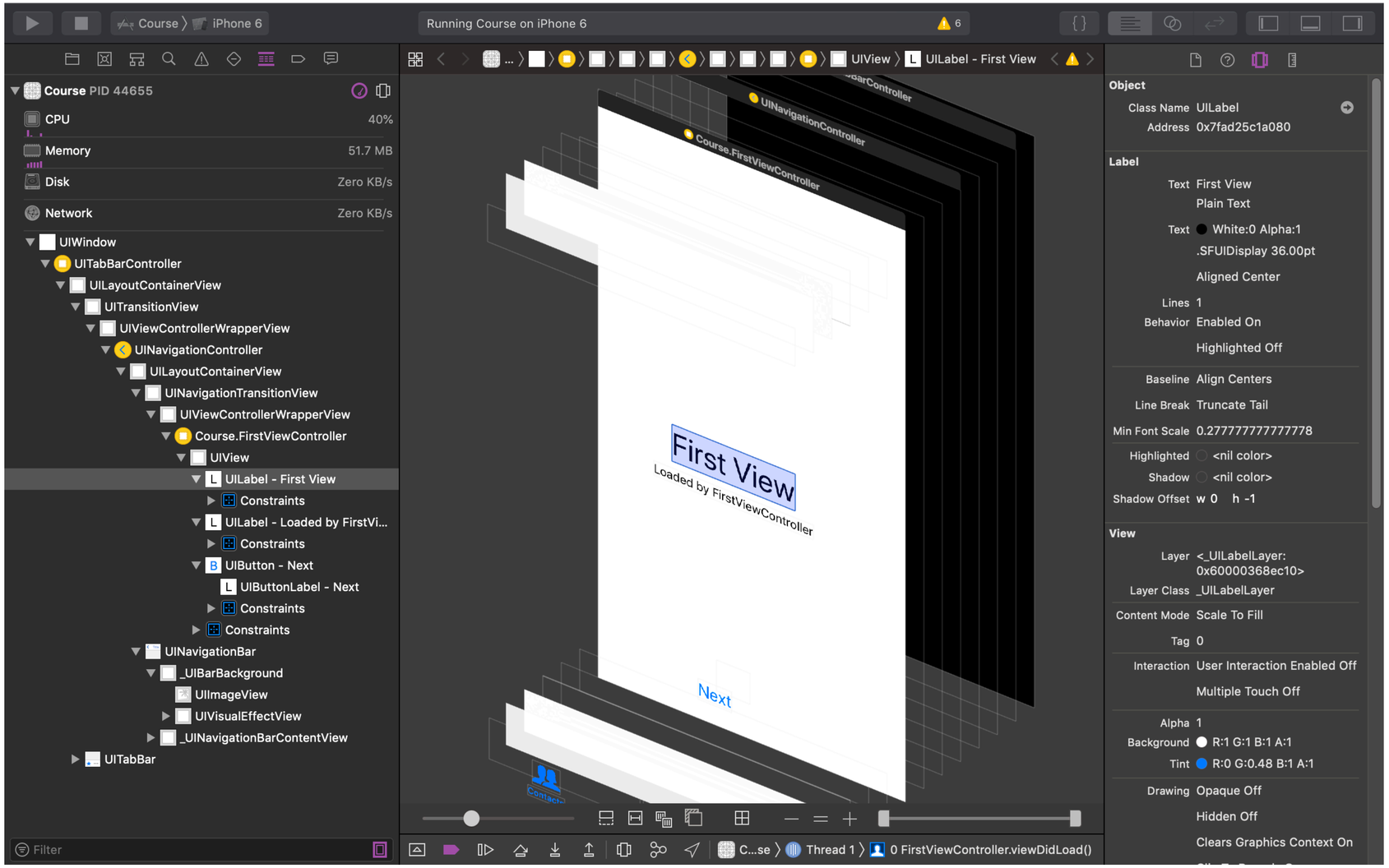
Edit & Custom breakpoints



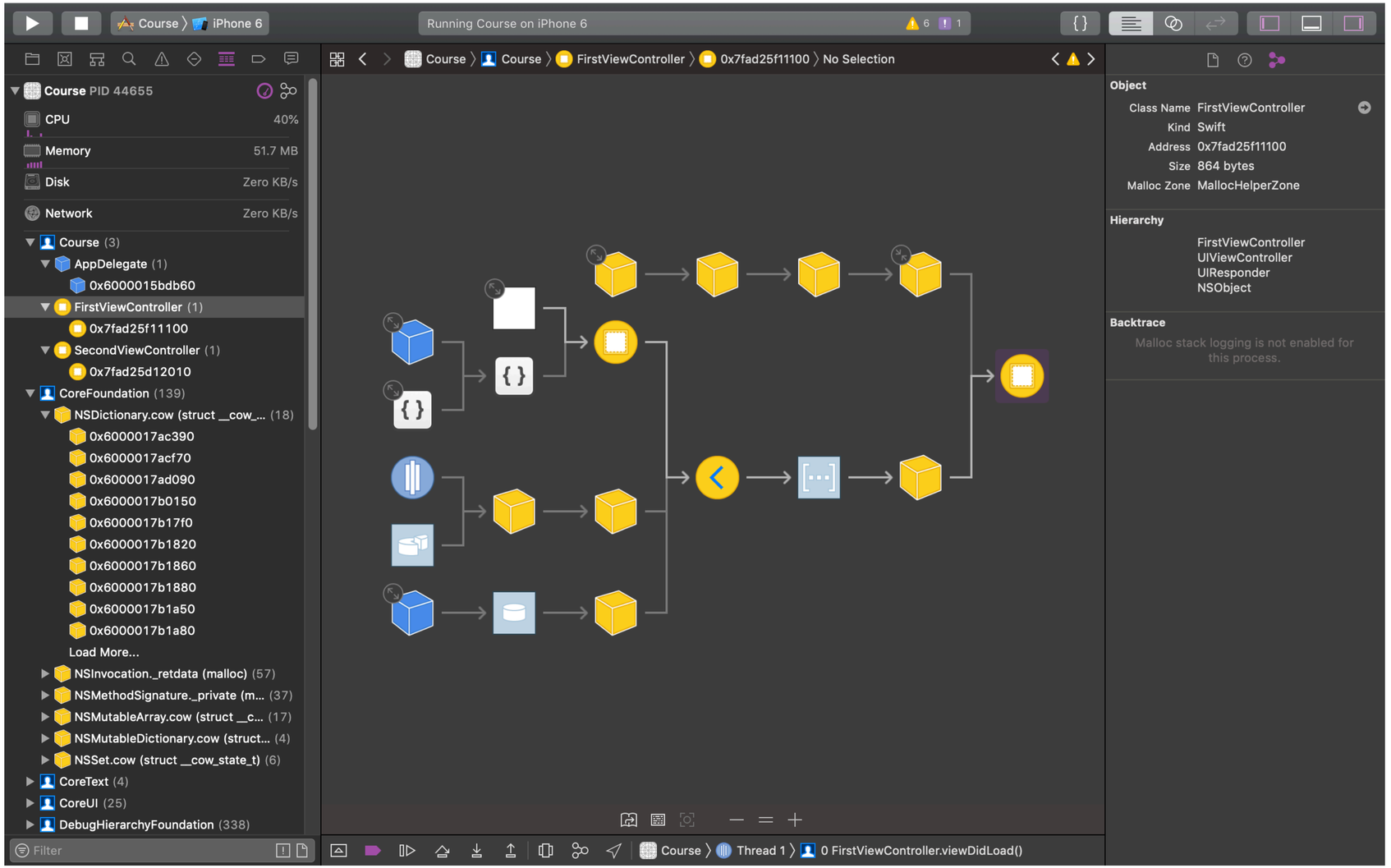




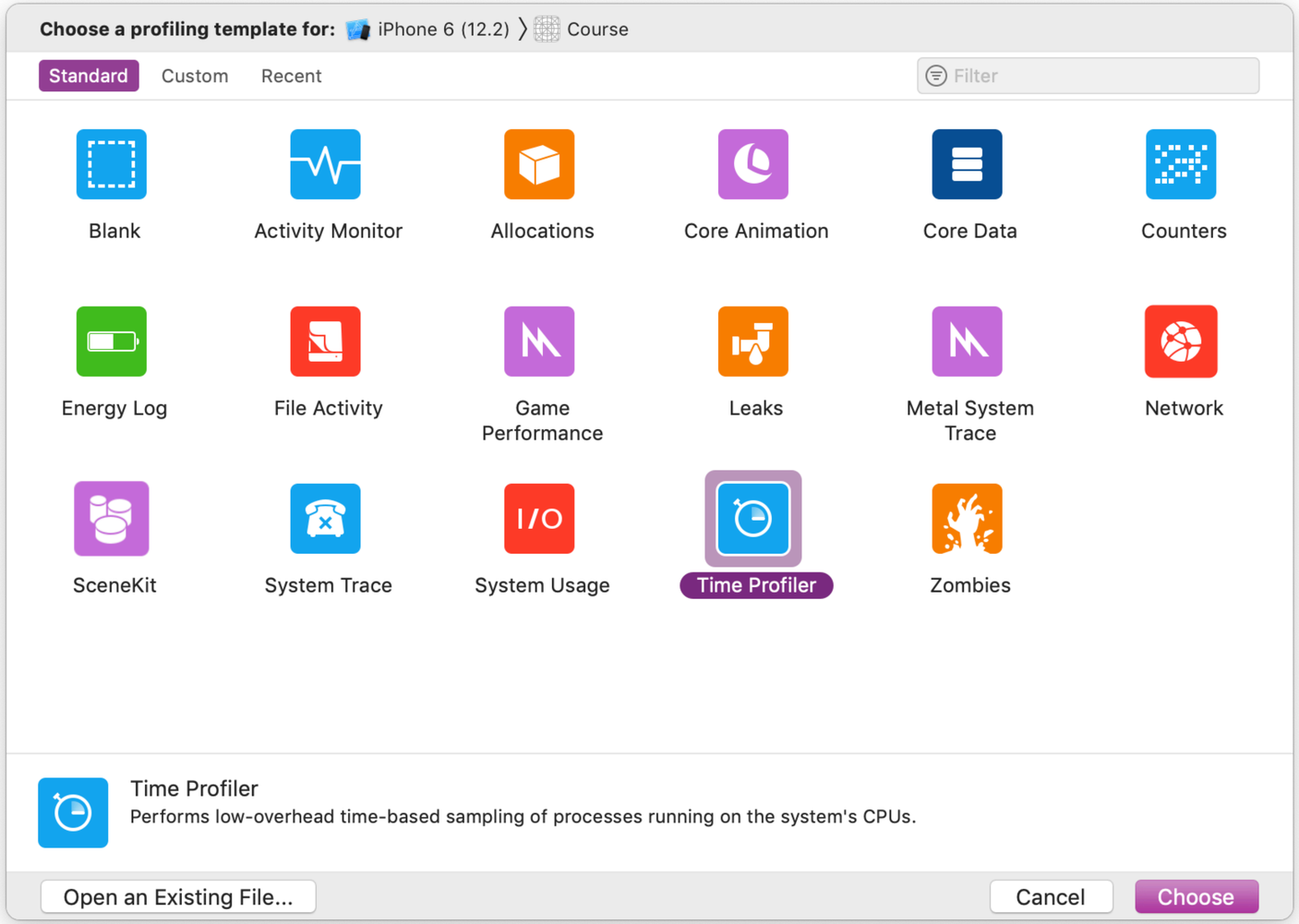
View debugging



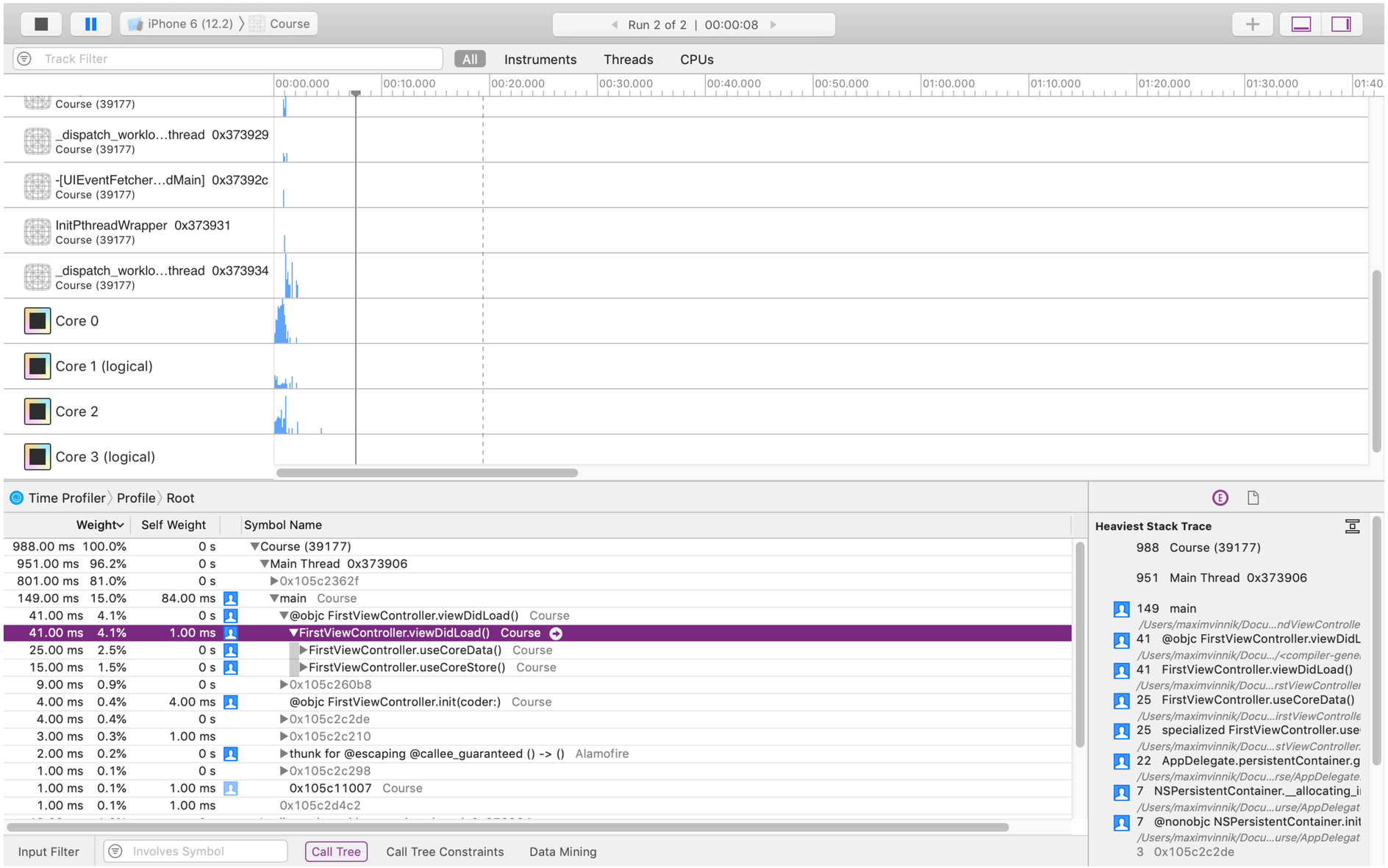
Memory debugging



Instruments

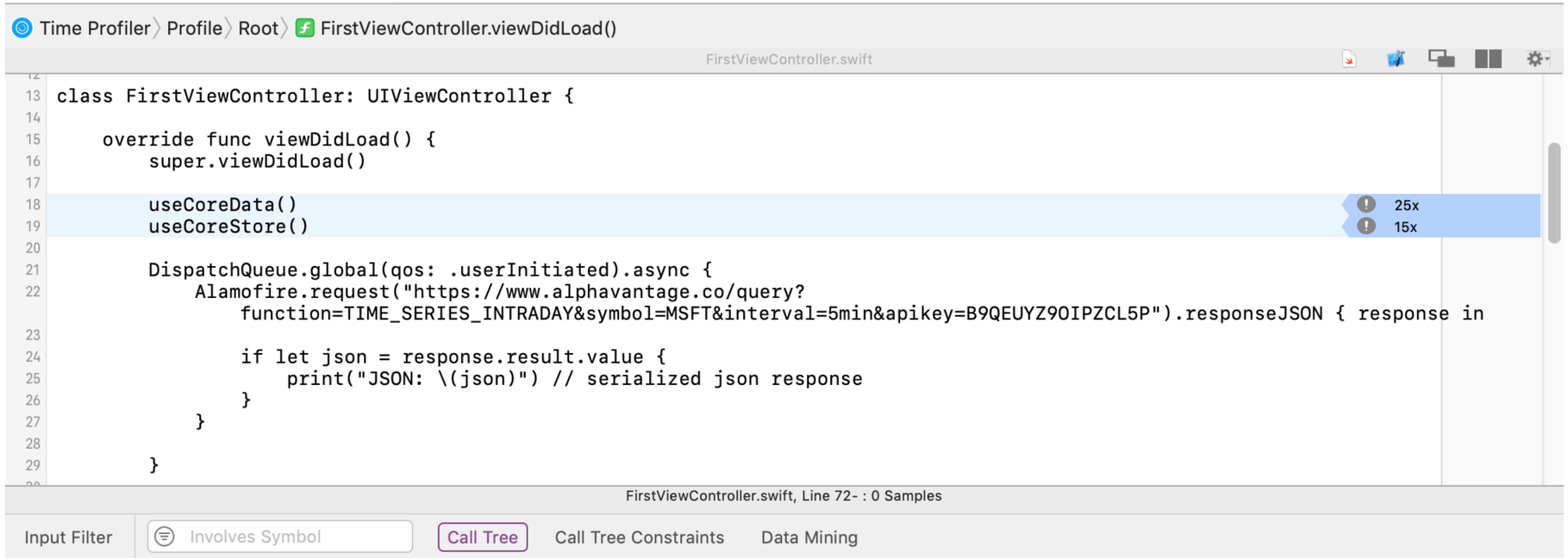


Time Profiler



View code in Instruments

By double clicking on some line in call tree, instruments will open corresponding code if it is available. Also some lines will be highlighted, showing some measured value.



Simulator Debugging

Simulator itself has some debug options you can turn on