

MOBGEN^o

Part of Accenture Digital

IOS TALKS

ADVANCE DEBUGGING

Change vars and visuals things with console

po button

```
<UIButton: 0x119d1ffd0; frame = (20 718; 335 40); opaque = NO; layer =  
<CALayer: 0x1d0038ea0>>
```

expression --object-description -- button

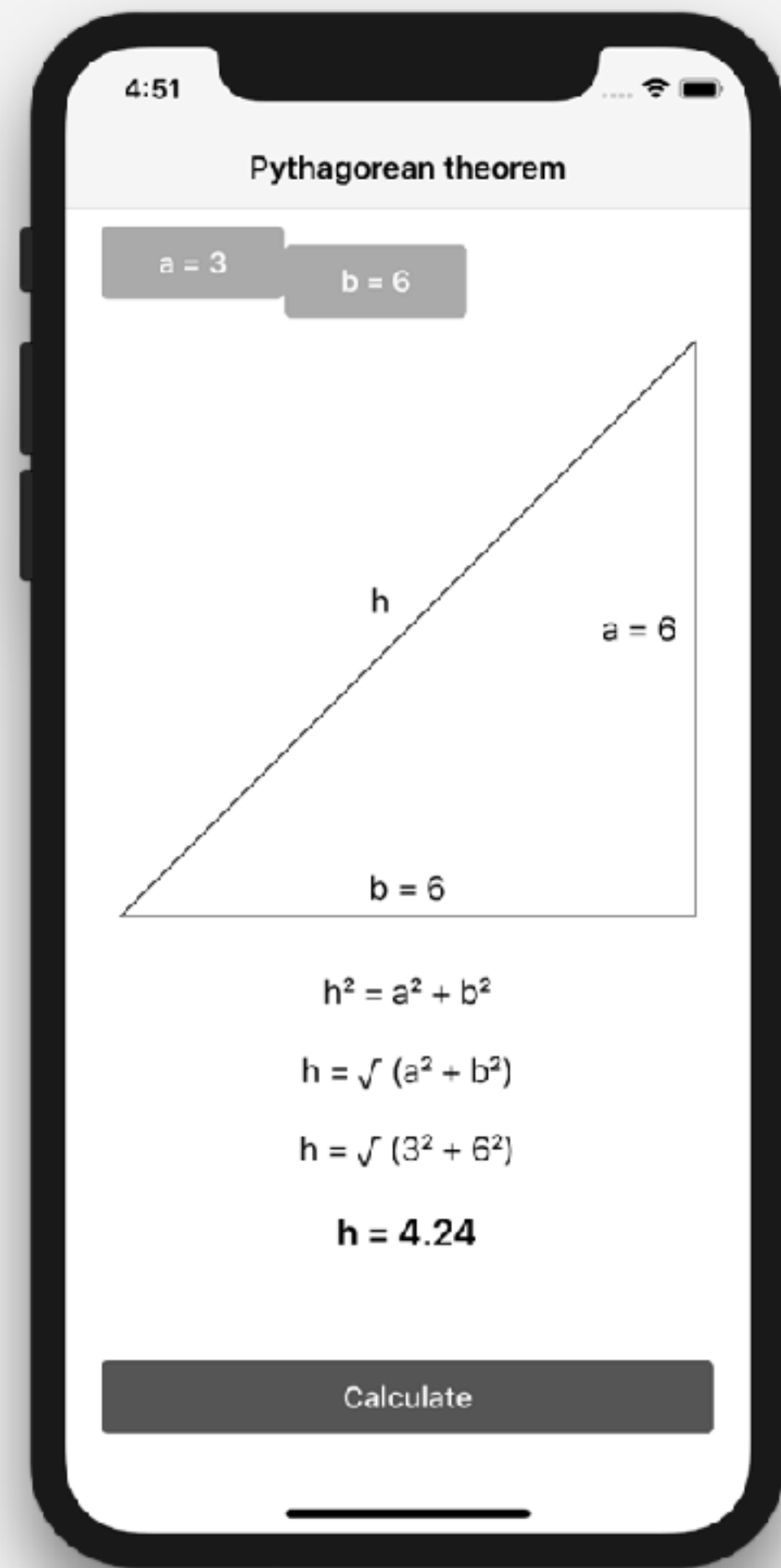
expression -O -- button

p button

expression button

```
(UIButton) $R1 = 0x00000000119d1ffd0 {  
    UIKit.UIControl = {  
        baseUIView@0 = {  
            UIResponder = {  
                NSObject = {  
                    isa = 0x00000000119d1ffd0  
                }  
            }  
        }  
        ....  
        _targetActions = 0x000000001d004add0 1 element  
        _downTime = 556621562.10544395  
    }  
}
```

Command	Alias For
expression	<expression>
p <expression>	expression -- <expression>
po <expression>	expression --object-description -- <expression>
* poc <expression>	expression --object-description -l objc -- <expression>



DEMO

ADVANCE DEBUGGING EXAMPLE

RESUME

poc [`self.view` recursiveDescription]

expression aButton.frame = CGRect(x: 10, y: 10, width: 100, height: 40)

flush

expression button.backgroundColor = .red

flush

expression aLabel?.text = String(format: "a = %.0f", aValue!)

thread jump --by 1

Watchpoints

ALIAS ALWAYS AVAILABLE

\$ vim ~/.lldbinit

command alias -h "Same as 'expression'" -- expr expression

command alias -h "Same as 'expression'" -- e expression

command alias -h "Refresh screen device to see the visual changes" -- flush expression -l objc -O -- [CATransaction flush]

command alias -h "Refresh screen device to see the visual changes" -- fl expression -l objc -O -- [CATransaction flush]

command alias -h "Same as 'po' but for swift with objc syntax" -- poc expression -l objc -O --