

Introdução

Swift é uma recente linguagem de programação desenvolvida com o principal objetivo de servir programadores que pretendem desenvolver aplicativos e programas para iOS, OS X, watchOS, tvOS (todos esses pertencentes a Apple) e Linux.

Exemplo de Código

Trecho do código do jogo FlappySwift, uma implementação de FlappyBird para iOS 8:

```
override func touchesBegan(touches: Set<UITouch>, withEvent event:
UIEvent?) {

    /* Called when a touch begins */

    if moving.speed > 0 {

        for touch: AnyObject in touches {

            let location = touch.locationInNode(self)

            bird.physicsBody?.velocity = CGVector(dx: 0, dy:
0)

            bird.physicsBody?.applyImpulse(CGVector(dx: 0, dy:
30)

        }

    } else if canRestart {

        Self.resetScene()

    }

}
```

Origens e Influências

O projeto de criação da linguagem Swift foi desenvolvido por programadores da Apple, liderados por Chris Lattner.

Teve início em 2010, sendo lançado em 2014 sua primeira versão oficial. E mais tarde recebendo versões em 2015 e em 2016. Desenvolvido principalmente como uma alternativa ao Objective-C, com uma escrita mais simples e um poder maior de expressão entre outras otimizações, apesar de possuir similaridades a C e Objective-C. Swift foi projetada para trabalhar com os frameworks da Apple, Cocoa e Cocoa Touch, e com códigos em Objective-C.

Foi influenciada por várias linguagens, principalmente: C, C#, Objective-C, D, Haskell, Python, Ruby e Rust. A seguir uma breve linha do tempo para ilustrar suas influências.

1972 - C

1983 - Objective-C

1990 - Haskell

1991 - Python

1995 - Ruby

2000 - C#

2001 - D

2010 - Rust

2014 - Swift

Classificação

Swift é uma linguagem compilada e multi - paradigma, sendo:

- Imperativa
- Funcional
- Orientada a objetos

Quanto a tipagem:

- Forte
- Inferida
- Estática

Avaliação Comparativa

- Leitura (readability):

Por possuir muitas classes, bibliotecas e paradigmas diferentes, a leitura de programas se torna mais difícil, embora isso dependa do programa em questão. Semelhante ao que acontece em Java. Por outro lado C e Python possuem uma leitura mais compreensível.

- Escrita (writability)

A escrita é mais dinâmica e fácil de ser desenvolvida como em Java e Python (possuem similaridades na parte orientada a objetos). A tipagem inferida e forte contribui para um código mais enxuto e fácil de escrever. Devido a restrições da linguagem, C tem uma escrita mais difícil em comparação com Swift.

- Expressividade

É uma linguagem com alto poder de expressão. É possível transcrever o código de outras linguagens para Swift sem alterar muito o código ou até simplificando, permitindo em uma leitura mais legível.

Exemplo de bubblesort em objective - C e em Swift:

```

-(NSArray *)bubbleSort:(NSMutableArray *)unsortedDataArray
{
    long count = unsortedDataArray.count;

    int i;

    bool swapped = TRUE;

    while (swapped){

        swapped = FALSE;

        for (i=1; i<count;i++)

        {

            if ([unsortedDataArray objectAtIndex:(i-1)] > [unsortedDataArray
objectAtIndex:i])

            {

                [unsortedDataArray exchangeObjectAtIndex:(i-1) withObjectAtIndex:i];

                swapped = TRUE;

            }

            //bubbleSortCount ++; //Increment the count everytime a switch is done, this
line is not required in the production implementation.
        }

    }

    return unsortedDataArray;
}

```

```

func bubbleSort(inout numbers: [int]) -> () {

```

```
Let numbersLength = numbers.count

for var l = 0; l < numbersLength; l++ {
    for var j = 1; j < numbersLength-l; j++ {
        if numbers[j - 1] > numbers[j] {
            var swap = numbers[j-1]
            numbers[j - 1] = numbers[j]
            numbers[j] = swap
        }
    }
}
}
```

Bibliografia

Wikipedia:

[https://en.wikipedia.org/wiki/Swift_\(programming_language\)](https://en.wikipedia.org/wiki/Swift_(programming_language))

Swift, site oficial:

<https://swift.org/>

Github, código do jogo completo:

<https://github.com/fullstackio/FlappySwift>