

Coleções e Mais Iteração



Há muita repetição no jogo de
Trivia

imprime pergunta
recebe resposta e a armazena
checa se está correta
se estiver, adiciona um ponto
se não estiver, xinga

Elimine essa repetição usando
um laço

Laços não são suficientes!

```
let p1 = "Qual a capital da Croácia?"  
let respostaP1 = "Zagreb"
```

Cada iteração acessa um **conjunto diferente de variáveis**

Arrays

Variáveis que contém múltiplas posições e podem ser indexadas

Podem guardar elementos de um mesmo tipo

Exs.: um array de inteiros, um array de strings, um array de arrays

```
// declaração
// Cria um array sem elementos.
// É necessário informar o tipo!
var perguntas:[String] = []
// Alternativa: var perguntas:Array<String> = []

perguntas.append("Qual a capital da Croácia?")
perguntas.append("Em que ano Swift 1.0 apareceu?")

// Qual número será impresso pela última linha?
let numElementos = perguntas.count
print(numElementos)
```



```
// declaração
// Cria um array com 3 Strings.
// Agora não precisamos dizer o tipo.
var perguntas = ["Qual a capital da Croácia?",
    "Em que ano Swift 1.0 apareceu?",
    "Qual o nome do 2o álbum do Vampire Weekend?"]

// A 1ª posição fica no índice 0, 2ª no índice 1, etc.
perguntas[0] = "Quem disse que 'Tá tranquilo, tá favorável'?"
perguntas[1] = "Quem é a irmã de Arya Stark?"

// Qual número será impresso pela última linha?
numElementos = perguntas.count
print(numElementos)
```

Leitura de uma posição de um array

```
var perguntas = ["Qual a capital da Croácia?",  
    "Em que ano Swift 1.0 apareceu?",  
    "Qual o nome do segundo álbum do Vampire Weekend?"]
```

```
// Qual a capital da Croácia?  
print(perguntas[0])  
// Em que ano Python 1.0 tornou-se disponível?  
print(perguntas[1])
```

```
var perguntas = ["Qual a capital da Etiópia?",  
    "Em que ano Swift 1.0 tornou-se disponível?",  
    "Qual o nome do 2o álbum do Vampire Weekend?"]
```

```
var perguntaAtual = 0
```

```
// perguntas.count == 3
```

```
while perguntaAtual < perguntas.count {  
    print(perguntas[perguntaAtual])  
    perguntaAtual = perguntaAtual + 1  
}
```

Algumas funções e propriedades úteis

`remove:at:`

`insert:at:`

`isEmpty`

`index:of:`

`prefix: and suffix:`

Crie um programa que, dado um array de números qualquer, devolve a soma desses números. Crie vários arrays com diferentes elementos para testar seu programa.

Crie um programa que, dado um array de números qualquer (use os do exercício anterior), devolve o maior elemento desse array.

Crie um programa que, dados dois arrays de números, devolve um novo array contendo apenas os elementos ímpares de cada um.

Crie um programa que, dado um array de números de ponto flutuante, **remove** dele todos aqueles cuja parte decimal é diferente de zero.

Crie um programa que, dados dois arrays $A1$ e $A2$ de números, constrói um array contendo apenas os elementos de $A1$ que não estão em $A2$.

Elimine a repetição de código do seu jogo de Trivia.

Adicione mais três perguntas a ele.
Deu muito trabalho?

Crie um programa que lê uma sequência de números do teclado, armazenando-os em um array. Quando o usuário digitar <ENTER> sem nenhum número, seu programa deve calcular a média de todos os números armazenados e imprimir esse valor.

Crie um programa que lê um inteiro N do teclado, **calcula todos os números primos** entre 1 e N e os guarda em um array. Ao final da execução, o programa apresenta todos, separados por vírgulas (mas sem vírgula depois do último!)

Crie um programa que lê um inteiro N do teclado e **calcula os fatoriais** de todos os números entre 1 e N e os guarda em um array. Ao final da execução, o programa apresenta todos, separados por vírgulas (mas sem vírgula depois do último!)

Construa um jogo de forca em
Swift

Caso não já tenha feito assim,
modifique seu jogo de força para
que, a cada erro, todas as partes
“enforcadas” anteriormente seja
mencionadas.