# O jogo da moeda

# Uma primeira versão

A primeira versão do jogo escreverá no ecrã "moeda" ou "coroa" de forma aleatória.

Assim, o jogador deverá pensar numa opção (cara ou coroa) e depois executar o programa, verificando se acertou ou não.

Em Python, o código que implementa este jogo é o seguinte:

import random

moeda = random.choice(["cara", "coroa"])

print (moeda)

Os valores aleatórios são gerados por uma biblioteca específica de Python chamada [random](https://docs.python.org/3/library/random.html), pelo que em primeiro lugar é necessário importar essa biblioteca na primeira instrução.

De seguida o programa cria uma primeira variável, chamada moeda, à qual atribui o valor de "cara" ou ou o valor de "coroa", de forma aleatória.

A função [random.choice](https://docs.python.org/3/library/random.html#functions-for-sequences)() seleciona um valor de uma sequência de valores. As [listas](https://www.w3schools.com/python/python_lists.asp) em Python são sequências de valores delimitadas por parêntesis retos []. Neste caso, a nossa lista tem apenas 2 valores: "cara" ou "coroa". Cada uma destas palavras está delimitada por aspas "". Esta é a forma de definir texto, que em linguagem de programação se denomina de [cadeia de caracteres (*string*)](https://www.w3schools.com/python/python_strings.asp). Iremos abordar esta estrutura mais à frente...

Finalmente, para que se visualize o resultado da escolha do computador utiliza-se a função [print](https://www.w3schools.com/python/ref_func_print.asp).

|  |  |
| --- | --- |
| **Balloon animal** | *Agora experimente o jogo que acabou de programar!* |

# Uma versão mais interativa

A primeira versão do jogo funciona, mas não é muito interativa.... Vamos agora acrescentar mais algum código de forma que esta nova versão consiga apresentar como resultado se o jogador acertou ou falhou.

Em Python, o código que implementa esta nova versão do jogo é o seguinte:

import random

# 1. selecionar "cara" ou "coroa" (jogada do computador)

moeda = random.choice(["cara", "coroa"])

# 2. Ler jogada do jogador

jogada = input("Cara ou coroa? ")

# 3. Mostrar a jogada do computador

print ("Eu joguei", moeda, ". E você?")

# 4. Verificar quem venceu

if moeda == jogada:

print ("ganhou! ")

else:

print ("perdeu...")

|  |  |
| --- | --- |
| **Balloon animal** | *Jogue esta nova versão do jogo.* |

|  |  |
| --- | --- |
| Head with gears | *Experimente escrever as suas jogadas com caracteres maiúsculos. O que acontece?*  *Uma pista:* [*código ASCII*](https://pt.wikipedia.org/wiki/ASCII) |

O algoritmo desta nova versão é o seguinte:

1. Selecionar "cara" ou "coroa" aleatoriamente (jogada do computador)
2. Ler jogada do jogador
3. Mostrar a jogada do computador
4. Verificar quem venceu

O primeiro passo do algoritmo é codificado de forma idêntica ao que foi realizado na versão anterior para selecionar o valor aleatório da jogada do computador.

O segundo passo do algoritmo torna o programa interativo ao pedir ao jogador a sua jogada. Chama-se a este passo **Input**. A variável jogada guarda a jogada do jogador, que é lida através da função [input](https://docs.python.org/3/library/functions.html#input)(). É possível colocar na chamada da função input um texto que sirva de contexto à leitura da jogada. Neste caso, foi colocado o texto: "Cara ou coroa? ".

O terceiro passo do algoritmo mostra a jogada do computador no ecrã. Aqui começa o **Output**.

O quarto e último passo do algoritmo verifica quem venceu. Para tal precisámos de uma estrutura de controlo condicional – um [if](https://docs.python.org/3/reference/compound_stmts.html#the-if-statement). Esta estrutura avalia uma expressão condicional ou predicado (uma expressão que devolve verdadeiro ou falso) e, se o resultado for verdadeiro, segue um caminho, print("ganhou!"), mas se for falso, segue o caminho alternativo, print("perdeu…”).

Repare que, neste caso, a expressão condicional ou predicado é a comparação de duas expressões (as variáveis jogada e moeda) e por isso utiliza-se o operador == (dois sinais = seguidos para distinguir do operador de atribuição que é o =). É com base nesta expressão condicional que apenas um dos blocos de código (definidos por indentação) é executado.

Finalmente, repare que há quatro linhas que se iniciam com o carácter cardinal #. Tratam-se de comentários escritos em língua portuguesa. Servem para tornar o programa mais legível e são completamente ignoradas pelo interpretador de Python.

|  |  |
| --- | --- |
| Eye | Repare que os comentários contêm os passos do algoritmo.  É sempre importante colocar comentários que descrevam de uma forma clara o algoritmo que foi delineado!  Identifique uma ou duas situações em que os comentários poderão ser úteis a quem escreveu o programa ou até a outras pessoas. |

|  |  |
| --- | --- |
| Eye | Apesar de se tratar de um programa muito simples, observe como ficaria muito limitada a sua legibilidade quando se retiram os comentários e não se escolhem nomes adequados para as variáveis:  import random  x = random.choice(["cara", "coroa"])  y = input("cara ou coroa? ")  print ("Eu joguei", moeda)  if x == y:  print ("ganhou! ")  else:  print ("perdeu...") |

# Desafio final

Tente agora adaptar o seu jogo ao lançamento de um dado (6 faces com números de 1 a 6).

**Qual será a condição de vitória?**

A condição de vitória é um dos componentes nucleares de um jogo e define-se através de uma (ou mais) expressão condicional que indica o vencedor de um jogo.

Com 6 valores distintos terá que decidir como se vence esta nova versão do jogo.

|  |  |
| --- | --- |
| **Share with person** | Partilhe o **seu** jogo! |