

# GABARITO AD2 - 1º SEMESTRE DE 2017.

Fundação CECIERJ - Vice Presidência de Educação Superior a Distância  
Curso de Tecnologia em Sistemas de Computação  
Disciplina: Programação com Interfaces Gráficas  
Professores: Mario Benevides e Paulo Roma

20 de maio de 2017

## Resumo

*Bulls and Cows* é um jogo de adivinhação de palavras. Há dois jogadores: Jogador 1 e Jogador 2. No início, é gerada uma palavra secreta e o Jogador 1 deve tentar adivinhá-la, fornecendo uma série de palavras. Quando o Jogador 1 fornecer uma palavra que não possua ao menos uma nova letra na posição correta, ele passa a vez ao Jogador 2. O vencedor é aquele que conseguir adivinhar a palavra secreta.

## 1 Implementação

Assim como descrito no enunciado da atividade, deve ser criada uma interface para o jogo desenvolvido na AD1.

A seguir, temos um possível modelo de uma boa interface para o jogo.

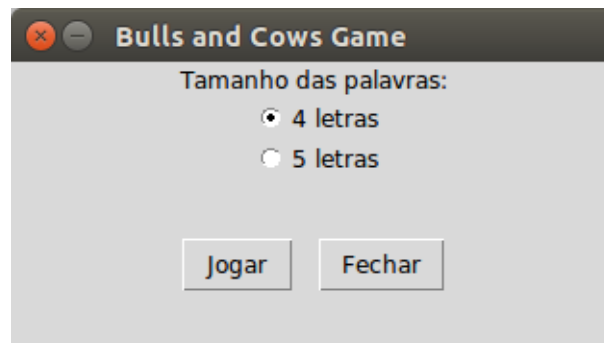


Figura 1: Tela inicial

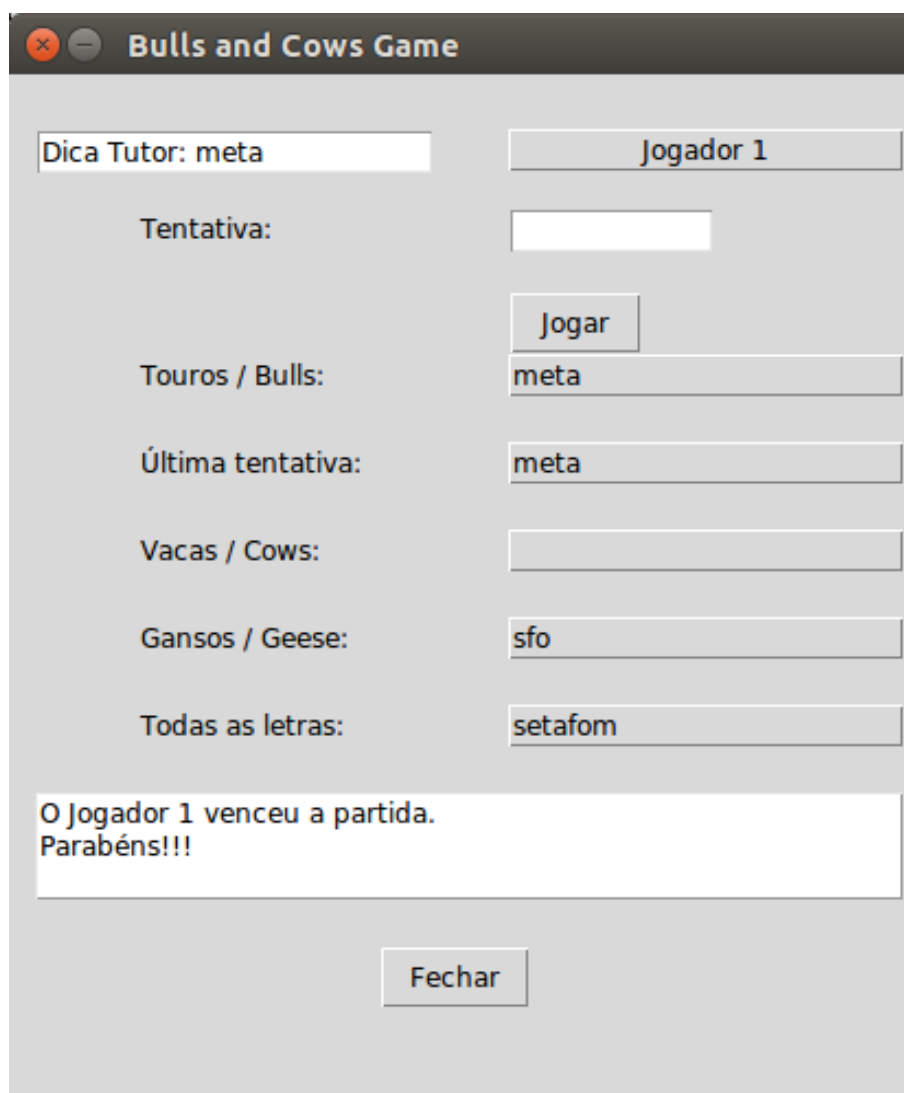


Figura 2: Tela do jogo.

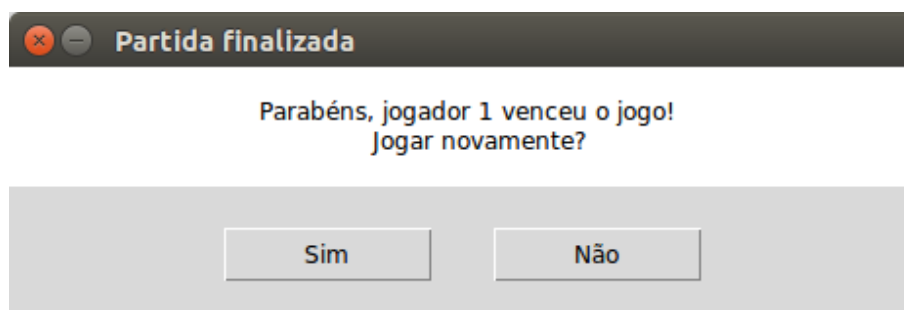


Figura 3: Tela final com mensagem de continuação.

Exemplo de comportamento da execução de **GraphUI.py**: O programa deve exibir uma tela com a opção de escolha do tamanho das palavras (Figura 1). A seguir, será apresentada uma tela (Figura 2) com a dica da palavra secreta, para que o Tutor possa testá-lo mais facilmente. Após a palavra ser adivinhada, o programa deve perguntar ao usuário, se deseja continuar jogando, ou se quer finalizar o programa (Figura 3).

## 2 Busca Binária

Exemplo da implementação do método de busca binária.

```
##
# Returns the index of a string in a sorted array,
# or -1 if the string is not present.
#
# @param alist
#   array of strings
# @param word
#   the string for which to search
# @return
#   index of the string, or -1 if it is not found
#
def binarySearch(self, alist, word):
    start = 0
    end = len(alist) - 1
    while start <= end:
        mid = (start + end) // 2
        if word == alist[mid]:
            return mid
        else:
            if word < alist[mid]:
                end = mid - 1
            else:
                start = mid + 1
    return -1
```

Lembre-se de que a correção será efetuada no ambiente Linux com codificação dos arquivos em utf8. Assim, sempre coloquem no cabeçalho dos seus programas, as duas linhas abaixo:

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
```