

# Trabalho 2 (2023/2)

## Disciplina de Programação Funcional

### A) Objetivo:

O objetivo deste trabalho é consolidar o conhecimento sobre construção de funções recursivas e programação interativa em Haskell.

### B) Enunciado:

O trabalho consiste na implementação de uma versão do “Jogo da Forca” via um programa de linha de comando (interação com o usuário via console de texto). A descrição do funcionamento essencial do “Jogo da Forca” pode ser encontrada em [https://pt.wikipedia.org/wiki/Jogo\\_da\\_forca](https://pt.wikipedia.org/wiki/Jogo_da_forca).

Para este trabalho, alguns requisitos foram elencados:

- Você deve implementar uma forma de desenhar o corpo do enforcado, além do número de tentativas que restam. Sugere-se desenhar diretamente com caracteres no estilo “ASCII Art” (veja mais informações em [https://pt.wikipedia.org/wiki/ASCII\\_art](https://pt.wikipedia.org/wiki/ASCII_art)).
- A palavra deve ser sorteada aleatoriamente a partir de um arquivo texto contendo um dicionário de palavras da língua portuguesa (no mínimo 25 palavras). Dicas: o módulo “System.Random” possui funções para geradores aleatórios, o módulo “System.IO” possui funções para entrada e saída de dados.
- Se o jogador estiver próximo da vitória (ou seja, a uma letra de acertar a palavra), o jogo pode decidir aleatoriamente (chance de 30%) sortear uma nova palavra. Neste caso, as letras que o usuário já acertou da palavra antiga devem preencher as posições da nova palavra de forma adequada.
- Ao final da partida, o jogo deve perguntar ao jogador se ele deseja jogar novamente ou não.
- **IMPORTANTE:** você deve obrigatoriamente criar uma nova regra de variação sobre a mecânica tradicional do Jogo da Forca e implementá-la.

#### DICA:

O pacote “random” deve estar instalado para que a dependência possa ser adicionada ao código. Será necessário que pelo menos o gerenciador de pacotes “Cabal” esteja disponível. Uma instalação completa do GHCUp deve ter instalado o Cabal. Utilize os seguintes comandos em um prompt de shell:

```
cabal install --lib random  
ghci -package random main.hs
```

### C) Desenvolvimento e avaliação do trabalho:

- O trabalho pode ser realizado individualmente ou em grupos de, no máximo, 4 alunos.
- Programas que não consigam ser executados receberão nota zero.
- Mensagens de erro apresentadas durante a execução do programa serão consideradas como erros de execução, e acarretarão descontos na nota do trabalho.
- Os trabalhos serão avaliados de acordo com critérios a serem estabelecidos pelo professor da disciplina, considerando o que é pedido no enunciado e o que foi realizado com sucesso de acordo com os princípios do paradigma de programação funcional.
- **Trabalhos copiados resultarão em nota zero para todos os alunos envolvidos.**

### D) Entrega do trabalho:

- Todos os arquivos-fonte necessários à execução do trabalho deverão ser empacotados em um único arquivo (.zip) e submetidos através do sistema Moodle até a data de entrega.
- Não serão aceitos trabalhos enviados por correio eletrônico.
- Não serão aceitos trabalhos enviados fora do prazo estabelecido.