

TRABALHO 5 – EXERCÍCIOS PRÁTICOS

Esta avaliação é individual.

Este trabalho corresponde a alínea Exercícios e Trabalhos do plano de ensino e como tal terá peso na avaliação da RA1.

Para que seu trabalho seja avaliado você deverá postar, no ambiente virtual de aprendizagem, na área reservada para este fim, dois links com a solução do seu trabalho. O primeiro destes links deve apontar para arquivos no ambiente repl.it¹, onde será possível executar sua solução. O segundo link deve apontar para arquivos contendo os mesmos códigos, entretanto este código deve estar hospedado no ambiente Github. **Links que apontem para códigos diferentes em ambientes diferentes serão provocarão o zeramento do trabalho.**

Todos os códigos enviados devem conter o enunciado que está sendo resolvido, na forma de comentário, em pelo menos um dos arquivos enviados. **Todos os arquivos de código devem conter, na primeira linha, em forma de comentário, o nome completo do aluno.**

Você deve ler todo este documento antes de começar e considerar o seguinte código de ética:

- I. Você poderá discutir todas as questões com seus colegas de classe, professores e amigos. Poderá também consultar os livros de referência da disciplina, livros na biblioteca, virtual ou não, e a internet de forma geral e abrangente nos idiomas que desejar. Contudo o trabalho é seu e deverá ser escrito por você. Cópia é plágio.

Para conseguir os pontos referentes a esta avaliação você deverá seguir as instruções apresentadas no item Enunciado deste documento.

OBJETIVO

Pesquisar e praticar. Pesquisar os conteúdos que irão complementar o material apresentado em sala ou nos livros sugeridos na ementa e praticar estes mesmos conceitos. Esta é uma oportunidade para aprimorar sua formação e se destacar profissionalmente.

METODOLOGIA UTILIZADA

Você pode e deve fazer uso das ferramentas disponíveis online para busca de informações. Pode e deve consultar seus colegas de classe e **pode mandar dúvidas para o professor até o dia anterior a data limite de entrega.** Observe as seguintes indicações:

- ao usar o Google, lembre-se de usar os comandos de busca para melhorar a qualidade dos resultados.
- faça sua pesquisa considerando apenas arquivos no formato pdf, disponíveis em instituições de ensino, em qualquer idioma, publicados nos últimos 5 anos;
- Caso seja necessário regidir algum texto para este Trabalho, o texto deverá ser escrito segundo as normas da ABNT. Use apenas as normas que fazem sentido para o trabalho de pesquisa que você está desenvolvendo. Preocupe-se com as fontes, espaçamentos, formato de parágrafos e citações.

Por fim, lembre-se que nenhum trabalho, exercício, ou pesquisa científica, ou acadêmica, admite qualquer tipo de plágio e que todos os conceitos que você trouxer para o seu trabalho deverão

¹ Alternativamente você pode usar: [Online Haskell Compiler \(tutorialspoint.com\)](https://tutorialspoint.com)

ser acompanhados da citação correta. Lembre-se também que todos os trabalhos enviados passarão por um sistema de avaliação de plágio e que **trabalhos contendo plágio serão zerados**.

ENUNCIADO

A seguir estão listados alguns problemas que você deverá resolver utilizando a linguagem Haskell.

Todos os problemas devem ser resolvidos com a criação de funções usando a linguagem Haskell. **E todas estas funções precisam ter seus tipos explicitados.** Você não pode usar a inferência de tipos e deve, sempre que possível, definir os seus próprios tipos, usando *type* ou *data*.

Antes de cada solução, deverá existir um comentário contendo o enunciado da questão que aquela função específica resolve. Para isso você DEVE COPIAR E COLAR o enunciado de cada questão na forma de comentário no código Haskell.

Como todas as questões serão resolvidas por funções. A prova de que a função resolve uma determinada questão deve ser feita por meio da chamada desta função no *main* do módulo que você criar para a solução dos problemas. **Lembre-se você precisa testar a função com casos positivos e negativos, sempre que for necessário.**

Se, por algum motivo, o serviço online usados para a execução do trabalho esteja indisponível você deverá anexar à sua resposta, um link para um documento online (pdf) contendo uma prova da indisponibilidade destes serviços. A prova pode ser a captura das telas de erro onde seja possível identificar data e hora da indisponibilidade. Neste caso, você deve enviar um link para outro serviço online equivalente com a sua solução.

Atenção, não é necessário postar links para o github e cada um dos exercicios a seguir deve estar em um arquivo separado no Repl.it. Ou seja, sua entrega constará de 4 links para arquivos do Repl.it.

1. Em Haskell temos um conjunto de tipos especiais: *Maybe*, *Nothing* e *Just* seu trabalho será criar no repl.it um código, comentado, que explique estes tipos, sua funcionalidade e que contenha quatro exemplos do seu uso. Com uma restrição: você deve evitar qualquer explicação que inclua o uso de *Monads*. Lembre-se entre estes exemplos, um deve ser de sua autoria e todos os outros precisam ter suas fontes citadas.
2. Escreva uma função chamada *idade* que usando pelo menos um tipo definido por você que receba o tempo de vida em segundos de uma determinada pessoa, o nome de um planeta e devolva a idade desta pessoa em anos caso ela tivesse vivido naquele planeta. Sabendo que o período orbital dos planetas é dado por:
 - a. Mercúrio: 0.2408467 anos terrestres;
 - b. Vênus: 0.61519726 anos terrestres;
 - c. Terra: 1.0 anos terrestre equivalente a 365.25 dias, ou 31.557.600 segundos;
 - d. Marte: 1.8808158 anos terrestres;
 - e. Jupiter: 11.862615 anos terrestres;
 - f. Saturno: 29.447498 anos terrestres;
 - g. Urano: 84.016846 anos terrestres;
 - h. Netuno: 164.79132 anos terrestres;

3. Dada uma coleção de números, implemente as funções *manter* e *descartar* que recebem a coleção e um predicado. Sempre que o predicado for verdadeiro quando aplicado a um item da coleção original este item deve ser mantido ou descartado. Lembre-se dos valores imutáveis. Você não pode usar as funções *filter* e *reject* já disponíveis no Prelude ou em qualquer outra biblioteca.
4. Um dos jogos infantis mais populares no planeta, chamado de Jogo Da Forca, consiste na escolha de letras do alfabeto latino para tentar encontrar uma palavra que esteja oculta. Você deve implementar uma versão deste jogo em Haskell considerando as seguintes restrições: você deverá usar caracteres ASCII para representar a forca, o enforcado, e os espaços selecionados para cada palavra; os símbolos usados nas palavras serão apenas os caracteres do alfabeto latino minúsculos entre a e z; o banco de dados de palavras deve conter, no mínimo 10 palavras de 6 letras ou mais; a cada tentativa a tela precisa ser limpa e redesenhada ([ANSI Escape Codes · GitHub](#)). No repl.it não é possível importar a biblioteca responsável pela geração de números randômicos, pode gerar as palavras e usá-las em sequência ou, se preferir, use algum fator do sistema, como os últimos dígitos da data corrente ([Dates and Times \(sourceforge.net\)](#))

RUBRICAS DE AVALIAÇÃO

As notas serão atribuídas segundo as seguintes regras:

- I. Todas as funções estão corretamente definidas, testadas e as saídas estão corretamente formatadas: **Nota = 10**;
- II. Todas as funções estão corretamente definidas e testadas mas, as saídas não estão corretamente formatadas: **Nota = 10 – (2,5 * Nº Erros)**;
- III. Não foi possível rodar todas as funções, ou o código enviado não atende o enunciado: **Nota = 0 (zero)**;
- IV. Programa não pode ser acessado pelo professor: **Nota = 0 (zero)**.

CUIDADOS QUE VOCÊ PRECISA TOMAR:

As regras de perda de ponto por entrega fora do prazo, constantes no plano de ensino, se aplicam a este trabalho.

Programas com códigos idênticos serão zerados.

Certifique-se que o seu código pode ser acessado por alguém além de você. Cada ambiente tem uma regra diferente de postar o código de forma que ele seja acessível. Para não ter problemas, depois que terminar e antes de postar os códigos. Acesse os links usando outra identidade no seu navegador *web* ou peça para algum colega acessar seus links. **Programas com código inacessível serão zerados.**