

TRABALHO 5 – LIST COMPREHENSION

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 você vai usar apenas um link do repl.it e outro do Github para todas as respostas as seguintes questões:

1. Usando *List Comprehension* escreva uma função, chamada *divisoresden*, que devolva uma lista dos divisores de um número dado.
2. Usando *List Comprehension* escreva uma função, chamada *contaCaractere*, que conte a ocorrência de um caractere específico, em uma *string* dada.
3. Usando *List Comprehension* escreva uma função, chamada *dobroNaoNegativo*, que devolve o dobro dos valores dos elementos não negativos da lista de inteiros dada.
4. Usando *List Comprehension* escreva uma função, chamada *pitagoras*, que devolva uma lista de triplas, não repetidas, contendo os lados dos triângulos retângulos possíveis de serem construídos por inteiros entre 1 e um número inteiro dado.
5. Números perfeitos são aqueles cuja soma dos seus divisores é igual ao próprio número. Usando *List Comprehension* escreva uma função, chamada *numerosPerfeitos*, que devolva uma lista contendo todos os números perfeitos menores que um número dado. Lembre-se que você já tem uma função que devolve uma lista dos divisores de um número dado.
6. Usando *List Comprehension* escreva uma função, chamada *produtoEscalar*, que devolva o produto escalar entre duas listas de inteiros. Lembre-se, existem as funções *fst*, *snd* e *zip* no *prelude* que podem ser úteis.
7. Usando *List Comprehension* escreva uma função, chamada *primeirosPrimos*, que devolva uma lista contendo os *n* primeiros números primos a partir do número 2.

8. Usando *List Comprehension* escreva uma função, chamada *paresOrdenados*, que devolva uma lista de par ordenados contendo uma potência de 2 e uma potência de 3 até um determinado número dado. Observe que estes números podem ser bem grandes.

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. Uma ou mais funções não foram criadas ou não rodaram corretamente: **Nota = 10 – (1,25 * N° Erros)**;
- III. Funções com nomes diferentes dos estabelecidos neste documento, mesmo que rodem acarretarão a perda de **0,25** ponto por função.
- IV. Não foi possível rodar todas as funções, ou o código enviado não atende o enunciado: **Nota = 0 (zero)**;
- V. 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.**