

Trabalho Prático 1

Enunciado

Escreva um programa em Haskell para manipular simbolicamente polinómios. Para isso deverá escolher uma representação adequada de polinómio e definir funções para:

- normalizar polinómios; ex.: a forma normal de $(0*x^2 + 2*y + 5*z + y + 7*y^2)$ é $(7*y^2 + 3*y + 5*z)$
- adicionar polinómios
- multiplicar polinómios
- calcular a derivada de um polinómio

O *output* deverá ser uma *string* representando o resultado em forma normal. O *input* pode ser feito diretamente na representação interna, no entanto, permitir o *input* como uma *string* e implementar uma função de *parsing* que transforma a *string* na representação interna será valorizado em 0.5 (em 4) valores.

Condições de Realização

Constituição dos Grupos: grupos de 2 estudantes, inscritos na mesma turma teórico-prática. Excecionalmente, e apenas em caso de necessidade, podem aceitar-se grupos de 3 elementos.

Escolha de grupos: os grupos deverão ser indicados na atividade disponibilizada para o efeito no Moodle. Numa 1ª fase, um dos elementos do grupo deverá fazer a escolha do grupo (um qualquer grupo da turma prática que frequentemente); numa 2ª fase, o 2º elemento juntar-se-á ao grupo.

Prazos: a escolha de grupos deverá ser realizada durante as próximas duas semanas. A entrega do trabalho (código-fonte e ficheiro *readme*) deverá ser realizada até ao final do dia **23 de outubro** de 2022, na atividade a disponibilizar para o efeito no Moodle; a demonstração do trabalho deve ser realizada na aula prática da semana de 24 de outubro de 2022.

Pesos das Avaliações: ver ficha da Unidade Curricular no SIGARRA.

Linguagens e Ferramentas: o trabalho deve ser desenvolvido em Haskell, devendo ser garantido o seu funcionamento em Windows e Linux, usando o ghci, versão 9.0.1. Caso seja necessária alguma configuração (para além da instalação padrão do *software*), isso deverá estar expresso no ficheiro README que deverá ainda incluir os passos necessário para configurar e/ou instalar as componentes necessárias (em Windows e Linux). A impossibilidade de testar o código desenvolvido resultará em penalizações na avaliação. Todo o código deverá ser devidamente comentado.

Avaliação

Cada grupo deve entregar um ficheiro README.pdf e o código-fonte, bem como realizar uma demonstração do trabalho. A submissão deve ser em formato ZIP, e o nome do ficheiro deve ser:

PFL_TP1_#GRUPO.zip

em que #GRUPO é a designação do grupo. Exemplo: PFL_TP1_G01_12.zip

O ficheiro ZIP deverá conter um ficheiro README.pdf e os ficheiros de código-fonte Haskell, os quais deverão ser **devidamente comentados**. Os ficheiros do código-fonte deverão estar ao mesmo nível que o README.pdf. O ficheiro ZIP não deverá conter nenhum diretório.

O ficheiro README.pdf deverá conter:

- Uma descrição (e justificação) da escolha de representação interna de polinómios;
- Uma breve descrição da estratégia de implementação de cada funcionalidade;
- Exemplos de utilização que permitam testar todas as funcionalidades do programa.