

Trabalho 02 - Gráficos no terminal

Objetivos

O objetivo principal do trabalho é avaliar a capacidade do aluno de aplicar o processo de projeto de programa visto em sala e desenvolver um programa no paradigma funcional que funcione corretamente e esteja bem escrito, o que inclui:

- Fazer uma análise apropriada;
- Definir e usar tipos de dados adequados;
- Criar especificações de funções com assinatura, propósito e exemplos;
- Empregar as técnicas de implementação discutidas em sala;
- Verificar o programa com a biblioteca `examples`;
- Revisar e melhorar o código.

Instruções

O trabalho é individual e deve ser entregue no classroom até às 23 horas e 59 minutos do dia 02/08/2024.

Cada aluno deve agendar a data da apresentação em uma planilha que será disponibilizada em breve.

Trabalhos com suspeitas de cópia serão zerados, portanto, não compartilhe sua solução e evite problemas desse tipo. Também serão zerados os trabalhos feitos em parte ou na sua totalidade por terceiros (chatgpt, amigo programador, colega de outra turma, veterano, entre outros).

Na implementação use apenas as funções presentes no material “Resumo da linguagem Racket”. Não use funções de alta ordem.

Descrição

O terminal de comandos têm resistido ao teste do tempo e continua sendo muito utilizado. Além de ser uma ferramenta essencial para os desenvolvedores, permite a criação de programas iterativos bastante elaborados. O interesse por desenvolver aplicações para o terminal tem aumentado e novas bibliotecas para diversas linguagens têm sido criadas. Veja por exemplos as bibliotecas [rich](#) para Python, [ratatui](#) para Rust e [pterm](#) para Go.

Nesse trabalho vamos criar um programa para o terminal que gera gráficos para facilitar a leitura de dados. Considere por exemplo a seguinte tabela, que registra a quantidade de sovetes vendidos de cada sabor.

Tipo	Quantidade
Uva	60
Groselha	100
Limão	40
Alho	0

Para facilitar a leitura dos dados dessa tabela, poderíamos exibir um gráfico de barras que mostra o percentual de cada sabor em relação ao total vendido:

Uva	[=====]	30%
Groselha	[=====]	50%
Limão	[=====]	20%
Alho	[]	0%

Note que o tamanho de cada barra é proporcional ao valor que ela representa. Note também que o alinhamento dos rótulo e porcentagens é importante.

O trabalho consiste em desenvolver um programa (`grafico.rkt`) que leia da entrada padrão uma tabela com duas colunas, a primeira com o item e a segunda com a quantidade, e exiba na saída padrão um gráfico no estilo do exemplo.

A entrada será dada a partir de um arquivo, onde cada linha corresponde a uma linha da tabela. Por exemplo, o arquivo da tabela anterior teria como conteúdo

Uva 60
Groselha 100
Limão 40
Alho 0

A entrada será feita com a função `port->lines`, que lê toda a entrada de uma vez e devolve uma lista de strings, cada string correspondendo a uma linha da entrada. A saída será feita com a função `display-lines`, que exibe cada string de uma lista em uma linha da saída padrão.

Você pode usar a seguinte função principal para fazer a entrada e saída:

```
(define (main)
  (define entrada (port->lines))
  (define tabela (... entrada))
  (define saida (gera-grafico tabela))
  (display-lines saida))
```

A função `gera-grafico` é o “coração” do programa, ela recebe como entrada uma representação da tabela e gera como resultado uma lista de strings, onde cada string corresponde a uma linha do gráfico.

Para executar o programa use o comando

```
racket grafico.rkt < tabela.txt
```

Onde o `tabela.txt` é o arquivo com o conteúdo da tabela. Note que o seu programa deve funcionar com qualquer arquivo de entrada que descreva uma tabela.

Em caso de dúvidas, estou a disposição.

Bom trabalho!