

Paradigmas de Programação

Introdução

O projeto é um traçador de raios, que tem como objetivo em renderizar uma imagem, dada a descrição de uma cena. O algoritmo almeja trazer o comportamento de um raio físico para o contexto computacional, desta forma, conseguindo simular como aquela imagem se pareceria na vida real.

Dificuldades e surpresas

O primeiro ponto foi garantir que as equações estavam corretas, em diversos momentos acabava gerando imagens super estranha, até que comecei a testar cada uma individualmente e encontrava todos os problemas com mais facilidade (ao contrário de programação orientada a objetos ou programação estruturada)

A paralelização do projeto foi super tranquila, ao ver o threadscope, todos os sparks conseguiam ser criados e terminavam no estado “converted” contudo, o consumo de memória simultâneo estava altíssimo e tive que remover o paralelismo do código para que conseguisse executar em meu computador, o ideal seria otimizar as funções para fizessem os calculos que precisam e deixem em lazy somente o que fosse necessário.

Como utilizar

Para executar, basta utilizar o comando “stack run”

Parâmetros:

- variável work na função main: você pode descrever uma lista de esferas personalizadas
 - cada esfera recebe um vetor posição e um raio
- função constante samplePerPixel no arquivo Utils.hs, que é diretamente ligado a qualidade do anti serrilhado da imagem, com mais amostrar por pixel, conseguimos ter uma imagem mais suave.
- função constante depth no arquivo Utils.hs, determina o máximo de rebatimentos que um raio pode ter, interfere na qualidade da iluminação.

Vídeo

<https://youtu.be/R32RvzSIZOg>

