

MAC 323 - Estruturas de Dados

Primeiro semestre de 2009

Protetores do espaço sideral V2.0 – Entrega: 30 de junho de 2009

Este EP está sendo dado em conjunto com a disciplina MAC211 – Laboratório de Programação. Dessa forma, ele pode ser feito em grupos com até 3 alunos.

O enunciado abaixo substitui o enunciado original. Caso você deseje, pode implementar o que foi pedido anteriormente, e terá um bônus em sua nota.

Seja S um conjunto com n pontos que se movimentam em um grande retângulo do plano. Existem vários tipos de pontos em S : as naves, os tiros, os asteroides e cristais. Você deverá armazenar os pontos de S em uma estrutura de dados a fim de decidir rapidamente quando ocorrerão colisões entre esses objetos, quando um deles sairá do retângulo, *etc.* Note que um objeto só poderá colidir com outro que esteja perto dele. Assim, se você armazenar os pontos em árvores de busca binária (uma indexada pela coordenada x e a outra pela coordenada y), e, além disso, ligar as folhas da árvore com fios (de forma que percorrendo os fios eu tenho a lista de pontos ordenada), poderá determinar rapidamente para cada ponto quais são os possíveis eventos em que ele pode estar envolvido. Note que os eventos podem gerar mudanças na ordem relativa dos pontos, remoção de pontos (um asteroide explode), ou inserção de novos pontos (a nave atira). Sua estrutura de dados deverá manter apenas os pontos no retângulo, e deverá estar sempre correta. Abaixo listamos os tipos de eventos que podem ocorrer e você terá de tratar em seu programa:

- 1) Um ponto p_1 de S é x -menor (ou y -menor) que um ponto p_2 no instante t_i , porém, é x -maior (y -maior) no instante t_{i+1} .
- 2) Um tiro da nave acerta um asteroide ou cristal. Você deve remover o tiro e o asteroide ou cristal das estruturas.
- 3) O campo de força da nave captura um cristal (mesmo caso anterior).
- 4) Dois asteroides ou cristais se tocam (basta alterar o sentido dos pontos).
- 5) Se o campo desintegrador da nave destroi um asteroide ou cristal, este deve ser removido das estruturas.
- 6) A nave atira. Isso acarreta um novo ponto no espaço, que deve ser incluído na estrutura de dados.
- 7) Se um tiro, asteroide ou cristal deixa o retângulo, esse deve ser removido das estruturas.

- 8) Se uma nave se choca contra um asteroide, ambos são removidos das estruturas e, se não for fim de jogo, a nave deve ser inserida novamente nas árvores (reinicializando-a em algum ponto “livre” do espaço).

Inicialmente devemos calcular o primeiro evento de cada objeto do jogo (nave, tiro, asteroides e cristais). Esse cálculo deve ser capaz de dizer quando (em que tempo) um determinado evento irá ocorrer. Antes disso a estrutura de dados representa corretamente o conjunto de pontos, e não precisa ser atualizada. Ao tratar o evento, você deverá calcular o próximo evento de cada objeto envolvido (se continuarem a existir), e verificar qual o próximo evento a ser tratado.

Para este EP, vamos elaborar alguns testes automatizados para a validação da ED utilizada para manipulação dos objetos na tela; os pontos que devem ser testados são:

- Robustez: a ED funciona corretamente com muitos pontos? E com poucos pontos (zero, por exemplo)?
- Correção: a ED é atualizada corretamente, para todos os eventos listados?

Portanto, a bateria de testes deve verificar a correção e robustez da sua implementação.