

Projeto Jogo da Vida

Laboratório de Programação - LP/2023 – prof. Eli Banks

Introdução

Em 1970, o cientista John Conway inventou um jogo que simula o processo de nascimento, sobrevivência, e morte. O jogo da vida é um autômato celular e foi criado de modo a reproduzir, através de regras simples, as alterações e mudanças em grupos de seres vivos, tendo aplicações em diversas áreas da ciência (ver: http://pt.wikipedia.org/wiki/Jogo_da_vida).

A ideia base é que um ser vivo necessita de outros seres vivos para sobreviver e procriar, mas um excesso de densidade populacional provoca a morte do ser vivo devido à escassez de comida. Os indivíduos vivem num mundo matricial e a geração seguinte é gerada a partir da geração anterior de acordo com as seguintes regras:

- **Reprodução:** Um ser vivo nasce numa célula vazia se essa célula vazia tiver exatamente 3 seres vivos vizinhos.
- **Sobrevivência:** Um ser vivo que tenha 2 ou 3 vizinhos sobrevive para a geração seguinte.
- **Morte por falta de comida:** Um ser vivo com 4 ou mais vizinhos morre porque fica sem comida.
- **Morte por solidão:** Um ser vivo com 0 ou 1 vizinhos morre de solidão.

A cada geração, as regras devem ser aplicadas para todos os seres vivos ao mesmo tempo (isto é no mesmo passo) para obtermos o próximo passo ou geração.

Objetivo do projeto

O objetivo do projeto é criar um programa em C para simular o jogo da vida. Os indivíduos vivem numa matriz e o programa deve gerar a geração seguinte a partir das regras previamente apresentadas. Cada posição da matriz é uma célula que pode ter um “O” (para representar um ser vivo) ou um ponto “.” para indicar “vazio” ou “morto”. Cada célula tem um máximo de 8 células vizinhas (que podem ser representadas pelo caracter “+”). Para simplificar, consideraremos que o mundo é plano (pois fica complicado definir que a última célula é vizinha da primeira em um mundo esférico).

Pequeno exemplo de uma matriz 14 x14, considerando apenas 4 gerações:

	00	01	02	03	04	05	06	07	08	09	10	11	12	13
00
01
02
03
04	O	O	O	O	O
05
06
07
08	O	O	O	O	O
09
10
11
12
13

geracao 1: 10 celulas vivas

	00	01	02	03	04	05	06	07	08	09	10	11	12	13
00
01
02
03	O	O	O
04	O	O	O
05	O	O	O
06
07	O	O	O
08	O	O	O
09	O	O	O
10
11
12
13

geracao 2: 18 celulas vivas

	00	01	02	03	04	05	06	07	08	09	10	11	12	13
00
01
02	O
03	O	.	O
04	O	.	.	O
05	O	.	O
06
07	O	.	O
08	O	.	.	O
09	O	.	O
10	O
11
12
13

geracao 3: 14 celulas vivas

	00	01	02	03	04	05	06	07	08	09	10	11	12	13
00
01
02	O
03	O	O	O
04	O	O	.	O	O
05
06
07
08	O	O	.	O	O
09	O	O	O
10	O
11
12
13

geracao 4: 16 celulas vivas

O programa deverá implementar as funções e estruturas de dados necessárias para a execução da simulação e para a interface com o usuário. Vamos usar um padrão de projeto de sistemas interativos para construir o programa (padrão MVC). Além disso, o programa deverá permitir o armazenamento em arquivo das configurações das gerações iniciais para possíveis futuras novas execuções.

Detalhamento dos Requisitos Funcionais do Jogo

Os seguintes requisitos deverão ser implementados durante o desenvolvimento do projeto:

Etapa 1:

- ◆ REQ01: O sistema deve permitir ao usuário a definição do tamanho do mundo (mínimo 10 e máximo 60) procurando sempre caber numa tela texto sem retorno de nova linha.
- ◆ REQ02: O sistema deve permitir ao usuário definir os seres vivos da primeira geração da simulação (isto é: definir as coordenadas de cada célula). Caso a célula já existe, dar a opção de excluí-la do mapa.
- ◆ REQ03: O sistema deverá permitir ao usuário: apresentar o mapa das células vivas atuais (apresentadas com o caracter 'O')

Etapa 2:

- ◆ REQ04: O sistema deverá ter a opção (sim/não) para que sejam também apresentadas as células vizinhas-mortas (apresentadas com o caracter '+').
- ◆ REQ05: O sistema deverá permitir ao usuário limpar todas as células do mundo (limpar totalmente o mapa)

Etapa 3:

- ◆ REQ06: O sistema deverá permitir ao usuário iniciar a simulação utilizando uma matriz auxiliar para criar a próxima geração.
- ◆ REQ07: O sistema deverá permitir ao usuário estipular uma determinada quantidade de gerações a serem simuladas, sempre identificando e apresentando-as na tela.
- ◆ REQ08: O sistema deverá permitir ao usuário definir a velocidade de sucessão das gerações. No caso de velocidade zero, então a simulação deverá ser passo-a-passo conforme acionado pelo operador, sempre visualizando as gerações que se sucedem.

Etapa 4:

- ◆ REQ09: O sistema deverá ser aprimorado substituindo a matriz auxiliar por uma lista simplesmente ligada usando alocação dinâmica de memória, o que dispensa o dimensionamento prévio de espaço para a memória auxiliar e otimiza todo o processo.
- ◆ REQ10: O sistema deverá apresentar na tela logo abaixo do mapa, as listas de células vivas e de vizinhos-mortos.

Etapa 5:

- ◆ REQ11: O sistema deverá permitir ao usuário salvar as gerações iniciais em arquivo, o qual deverá conter todas as gerações iniciais que o usuário quiser salvar, gravadas sucessivamente.
- ◆ REQ12: O sistema deverá permitir ao usuário carregar cada geração previamente armazenada em arquivo de forma sucessiva conforme seu comando.

Etapa 6:

- ◆ REQ13: Preparação do relatório final para entrega junto ao Projeto em sua fase final.

Estrutura de Dados do Jogos

O mapa (ou mundo) que representa a estrutura matricial dos indivíduos deverá ser implementada usando tipo agregado de homogêneo (matriz) e cada indivíduo deverá ser devidamente representado com as informações necessárias para o contexto do problema. Sugere-se inicialmente criar duas matrizes bidimensionais: uma original e outra auxiliar na qual estarão representados cada célula.

Interface com o Usuário

A interface com o usuário deverá ser no modo texto (terminal) e sempre a mais elucidativa possível.

Simuladores OnLine do Jogo da Vida

- <http://web.stanford.edu/~cdebs/GameOfLife/>
- <https://bitstorm.org/gameoflife/>

Questões de Ordem.

- O programa deverá ser desenvolvido na linguagem C (padrão ANSI) utilizando a técnica MVC, conforme o projeto anterior.
- O código desenvolvido deverá estar bem estruturado e com o uso correto de funções
- A interface do jogo deve ser no modo texto (terminal)
- Os grupos deverão ser os mesmos do projeto anterior. Será descontado 0,5 ponto por dia atraso na formação dos grupos.
- Durante as aulas de LP as equipes deverão trabalhar no projeto e tirar dúvidas sobre os conceitos de LP e construtores da linguagem C necessários para o desenvolvimento do projeto
- Cada etapa definida acima deverá ser entregue no formato . **zip**, no prazo determinado e valerá nota para o grupo.
- Não serão aceitos trabalhos entregues não . zip, fora do prazo ou através de e-mail
- O projeto deverá ser desenvolvido em equipe de no mínimo 3 e no máximo 4 alunos
- Todos os alunos do grupo deverão estar presentes no dia da apresentação final
- Trabalhos copiados ou com indícios de cópia receberão nota zero!
- Código totalmente ou parcialmente copiado da Internet serão zerados