

Introdução ao desenvolvimento do motor do Jogo

1. O que é um jogo de Computador

Um jogo de computador é um aplicativo que em muitos aspectos se assemelha a qualquer outro programa. Entretanto, especialmente jogos 3D, possui as seguintes características:

- Deve executar em tempo real: Isso significa que o tempo de resposta de uma ação do jogador deve ser mínimo, não comprometendo a interatividade. Neste contexto, tempo real significa atualizar a tela com uma taxa de no mínimo 20 fps (*frames per second*). Quanto maior for o frame rate, melhor é a interatividade;
- Faz uso de técnicas de Computação Gráfica (CG) em tempo real. Neste caso, são usadas APIs gráficas, como OpenGL e DirectX, para garantir a renderização rápida das cenas. Para isso, é necessário que o computador possua uma placa de vídeo aceleradora;
- Deve implementar algoritmos de IA para o gerenciamento de personagens autônomos, que é usada para atender aos seguintes requisitos
 - **Percepção do mundo:** Os personagens devem perceber o ambiente onde estão inseridos para poder tomar decisões apropriadas a cada instante.
 - **Tomada de decisões:** consiste em escolher a melhor ação a ser realizada frente a situação atual do mundo;
 - **Planejamento:** Personagens mais avançados podem fazer um planejamento de ações de alto nível, deixando as decisões mais rotineiras para o módulo de tomada de decisões;
 - **Aprendizado:** Alguns personagens podem aprender com o tempo, especialmente frente a erros cometidos;
 - **Comportamentos:** representa tudo que o personagem pode realizar, como andar, conversar, realizar ações, etc;
 - **Escolha de caminhos:** O personagem deve possuir recursos para fazer um planejamento de caminho a baixo custo para ir de um local para outro.
- Pode implementar recursos de comunicação em rede, caso o jogo for multi-player;
- Deve disponibilizar interfaces por onde o usuário interage com o jogo;
- Pode fazer uso de simulações físicas para a representação de comportamentos e ações mais realistas.

2. Algoritmo genérico de um jogo 3D em Tempo Real

Toda aplicação gráfica em tempo real tem uma estrutura de execução muito similar: existe um laço de repetição infinito

```
While(true)
{
    Para o personagem sob controle do jogador
    {
        Avalia entrada do usuário via teclado, mouse, joystick
        Avalia mensagens recebidas
        Executa efeito da entrada (ação) e mensagens sobre o personagem
        Enviar possíveis mensagens
    }
}
```

```

Para cada personagem autônomo
{
    Avalia mensagens recebidas
    Faz avaliação do mundo (percepção)
    Faz processamento da IA
    Determina ação a realizar
    Executa ação e efeitos das mensagens
    Envia possíveis mensagens para outros atores
}
Processar física do jogo
Renderizar cenário e todos os personagens
Processar áudio
}

```

Nesta implementação, a cada loop de execução todos os personagens autônomos fazem processamento de suas ações, podendo enviar e/ou receber mensagens de outros personagens.

Questões:

- Qual deve ser a ordem de execução das ações? Ou seja, se 2 personagens dispararem contra o outro, dependendo da ordem de execução das ações, um pode morrer e o outro nem se ferir.
- O que são as mensagens, qual o conteúdo, como são implementadas e para que servem?
- Implementação da IA dos personagens via threads ou via laço de repetição?
- O processamento da física, IA e renderização deve ser feito a cada loop, ou seja, eram processados em função de um único framerate. Atualmente, jogos mais modernos adotam uma taxa única para cada componente. Por exemplo, a física pode ser atualizada 30x/s, a IA 10 e a renderização tão rápida quanto a máquina permitir [Buckland, pg 29].

3. Engines

Para o rápido desenvolvimento do aplicativo do jogo (programa), geralmente faz-se uso de engines, também chamados motores. Os engines gráficos são os mais utilizados e mais disseminados. Esses motores disponibilizam uma série de funcionalidades que podem ser utilizadas pelos programadores de jogos. Isso reduz os esforços de programação e assegura a geração de um programa com melhor qualidade. O principal aspecto tratado pelo engine refere-se a algoritmos gráficos e interação com a placa gráfica. Atualmente, existem no mercado centenas de engines, que geralmente são desenvolvidos para atender a tipos específicos de jogos. Também existem engines que tratam a parte de simulação física do jogo. Entretanto, o desenvolvimento de engines para IA ainda é um grande desafio, visto que se está entrando em uma área cognitiva, cuja implementação depende muito do propósito do jogo. Ao contrário da CG, onde tudo se resume a processamento de triângulos, na IA é muito maior a diversidade de informação, que dificilmente poderá ser generalizada em uma única primitiva comum.

Para uma lista completa de engines já desenvolvidos, consulte <http://www.devmaster.net/engines/>