Centro Universitário Barão de Mauá

Estrutura de Dados

Trabalho I

Os RPGs, ou jogos de interpretação de personagens, são um dos gêneros mais populares. A narrativa clássica desses jogos acompanha um herói protagonista em uma jornada épica, repleta de obstáculos. Ao longo dessa empreitada, o herói enfrenta criaturas perigosas e adversários formidáveis. Para superar esses desafios, ele coleta itens valiosos, como armas e poções, que potencializam suas habilidades e atributos (vida, ataque, defesa, etc.). A dificuldade do jogo aumenta progressivamente. A cada fase, o herói e seus inimigos ficam mais fortes, e os desafios se tornam cada vez mais maiores.

Para o Trabalho I você deverá implementar um RPG básico. As especificações técnicas, incluindo componentes, regras e interface do usuário, são apresentadas nas próximas seções.

Componentes do Jogo:

Níveis: O jogo possuirá um número de níveis, maior do que um, que pode ser fixo ou aleatório (decidido no início da aplicação).

Mapa: Cada nível possuirá um caminho. O tamanho do caminho pode ser fixo ou aleatório (decidido no início do nível). Um caminho estará constituído de sqms, cada um dos quais podendo estar vazio, ter um elemento (ver descrição mais a frente) ou um inimigo (ver descrição mais a frente). O último sqm do mapa indica o final do nível.

Elementos: existem, pelo menos, dois tipos de elementos: armas e poções. Todo elemento possui um nome e peso. Além disso, toda arma tem uma capacidade de ataque (um número), e toda poção uma capacidade de cura (um número de pontos de vida).

Inimigos: todo inimigo possui um nome, uma capacidade de ataque e um número de pontos de vida. Os valores dessas duas últimas variáveis podem ser aleatórios.

Herói: O herói possui um nome, uma quantidade de pontos de vida e dois objetos: um cinto e uma mochila. No cinto, o herói pode colocar elementos em qualquer posição, assim como acessá-los à vontade. Na mochila, o herói só pode colocar um objeto sobre o outro e, em todo momento, só pode acessar o elemento que está sobre os demais. Para poder atingir os elementos mais ao fundo, deverá retirar (e perder) os elementos anteriores. Cada um desses espaços tem uma capacidade máxima de peso de elemento. Por outro lado, a mochila não possui um limite de espaço nem peso.

Regras do jogo:

Percurso no mapa: o jogador avança de sqm em sqm até atingir o final do mapa.

Interação com elementos: Quando o herói chega numa sqm com um elemento, ele pode decidir pegá-lo ou descarta-lo. No primeiro caso, ele deve indicar se deseja coloca-lo na mochila ou no cinto. No segundo caso, o objeto desaparece do caminho.

Batalhas: Quando o herói chega num sqm com um inimigo, começa uma batalha. O jogador pode escolher qualquer arma que tenha no cinto ou na mochila (considerando as limitações de acesso a cada uma delas) para utilizá-la durante a batalha. Por turnos, o jogador e o inimigo atacam (os pontos de vida são diminuídos considerando a capacidade de ataque da arma ou do inimigo) até que um deles perca o total de pontos de vida. Se o jogador ganha a batalha, ele pode continuar no mapa; senão, o jogo termina.

Uso de poções: As poções permitem ao jogador recuperar alguns pontos de vida. Quando o jogador decide usar uma poção, os pontos de vida dela são passados ao jogador até atingir a sua capacidade total máxima. Depois disso, a poção desaparece.

Movimentação: Cada vez que o jogador chega num sqm, o jogo deve indicar o que ele encontrou ali e permitir o tipo de interação correspondente (se for o caso), seguindo as seguintes regras:

- Quando o jogador estiver em um sqm vazio, ele poderá manipular o conteúdo do cinto ou da mochila. Isto é, ele poderá retirar/usar elementos de qualquer um desses objetos. Além disso, ele poderá avançar ao próximo sqm.

- Quando o jogador estiver em um sqm com um elemento, pelo menos o processo de "interação com elemento" deverá ser realizado. Depois disso, ele passa automaticamente ao próximo sqm.

- Quando o jogador estiver em um sqm com um inimigo, pelo menos o processo de "batalha" deverá ser realizado. Depois disso, ele passa automaticamente ao próximo sqm.

Fim do jogo: O jogo pode terminar quando o jogador perde uma batalha ou quando ele atinge o final do último nível.

Importante: listas e uma pilha (pelo menos) devem ser usadas.

No início do jogo, a aplicação deve solicitar o nome do jogador.

Em todo momento, deve-se apresentar o conteúdo total do cinto e o elemento na parte de cima da mochila. Também os pontos de vida restantes do jogador.

Em todo momento, dependendo do tipo de sqm em que o jogador esteja, as opções de movimentação ou interação disponíveis devem ser apresentadas.

No fim do jogo, o nome e o nível máximo atingido pelo jogador devem ser armazenados no arquivo *high_scores.txt*.

O número de regras ou funcionalidades pode ser **incrementado** se o grupo desejar, sempre cumpra com, pelo menos, as regras/funcionalidades indicadas neste documento.

Atenção!

- Serão permitidos grupos de, no máximo, 3 alunos e deverão ser definidos e informados na SAV da disciplina até 23:58h do dia 06/09/2024. Após esta data não será permitida a mudança de grupos. Casos omissos serão tratados pelo docente.

- **Entrega: até as 23:58h do dia 29/09/2024 contendo todos os códigos fontes devidamente documentados, via Portal (SAV da disciplina ED);**
- **Não serão aceitos trabalhos**:
    - após a data e horário de entrega;
    - que utilizem códigos em outra linguagem de programação, senão C++;
    - que utilizem *containers* (*vector, queue, stack, priority_queue, list, set, map...*), bem como declarações de variáveis/ponteiros com *auto type*.
    - que utilizem outras bibliotecas/repositórios exceto a biblioteca padrão de C++ (https://en.cppreference.com/w/cpp/header).
- O código deve ser organizado usando divisão das classes entre arquivos de cabeçalho (.h) e implementação (.cpp);
- Obviamente seu programa deverá ser bem documentado, e a forma de utilizá-lo também será avaliada.
- Os programas resultantes do trabalho poderão serem testados na presença do professor durante o horário da aula de ED seguinte à data de entrega;
- **Todo arquivo de código fonte deve conter,** nas suas primeiras linhas, um campo de comentário com o **nome e o número institucional** dos responsáveis pelo trabalho;
- Trabalhos reconhecidos como 'muito semelhantes' pela sua estrutura de programação serão desconsiderados. Lembrem-se, variáveis com nomes diferentes, mas em códigos com a mesma estrutura, são considerados 'muito semelhantes';
- Ressalto que este trabalho pode envolver alguns conhecimentos da linguagem de programação C++ que não foram cobertos pelos exemplos ou pelos exercícios realizados em aula. No entanto, estes conhecimentos estão disponíveis nos livros referenciados como material de apoio na ementa da disciplina.

Bom trabalho!