

Descrição:

“Após matar o rei de Hyrule, o mago Agahnim está mantendo a princesa Zelda prisioneira e pretende romper o lacre que mantém o malvado Ganon aprisionado no Dark World. Link é o único guerreiro capaz de vencer o mago Agahnim, salvar a princesa Zelda e trazer a paz para o reino de Hyrule. Porém, a única arma forte o suficiente para derrotar o mago Agahnim é a lendária **Master Sword** (Figura 1), que encontra-se presa em um pedestal em **Lost Woods**. Para provar que é digno de empunhar a Master Sword, Link deve encontrar e reunir os três **Pingentes da Virtude**: coragem, poder e sabedoria (Figura 2). Os três pingentes encontram-se espalhados pelo reino de Hyrule, dentro de perigosas **Dungeons**.



Figura 1 – Master Sword

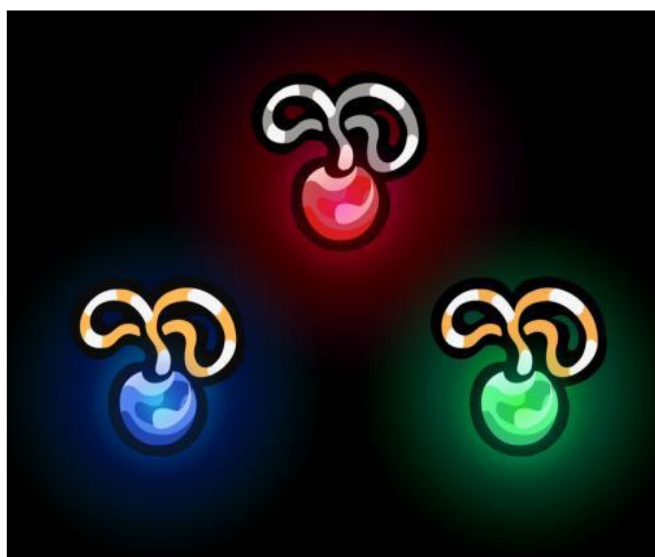


Figura 2 – Pingentes da Virtude

O Trabalho consiste em implementar um agente capaz de locomover-se autonomamente pelo reino de Hyrule, explorar as perigosas dungeons e reunir os três Pingentes da Virtude. Para isso, você deve utilizar o **algoritmo de busca heurística A***. O agente deve ser capaz de calcular automaticamente **a melhor rota** para **reunir os três pingentes da virtude e ir para Lost Woods**, onde está localizada a Master Sword. O mapa do reino de Hyrule é mostrado na Figura 3.

O reino de Hyrule é formado por 5 tipos de terrenos: grama (região verde claro), água (região azul), montanha (região marrom), areia (região marrom claro) e floresta (região verde escuro). Os custos para se passar por cada tipo de terreno são os seguintes:

- ☐ **Grama** – Custo: +10
- ☐ **Areia** – Custo: +20
- ☐ **Floresta** – Custo: +100
- ☐ **Montanha** – Custo: +150
- ☐ **Água** – Custo: +180

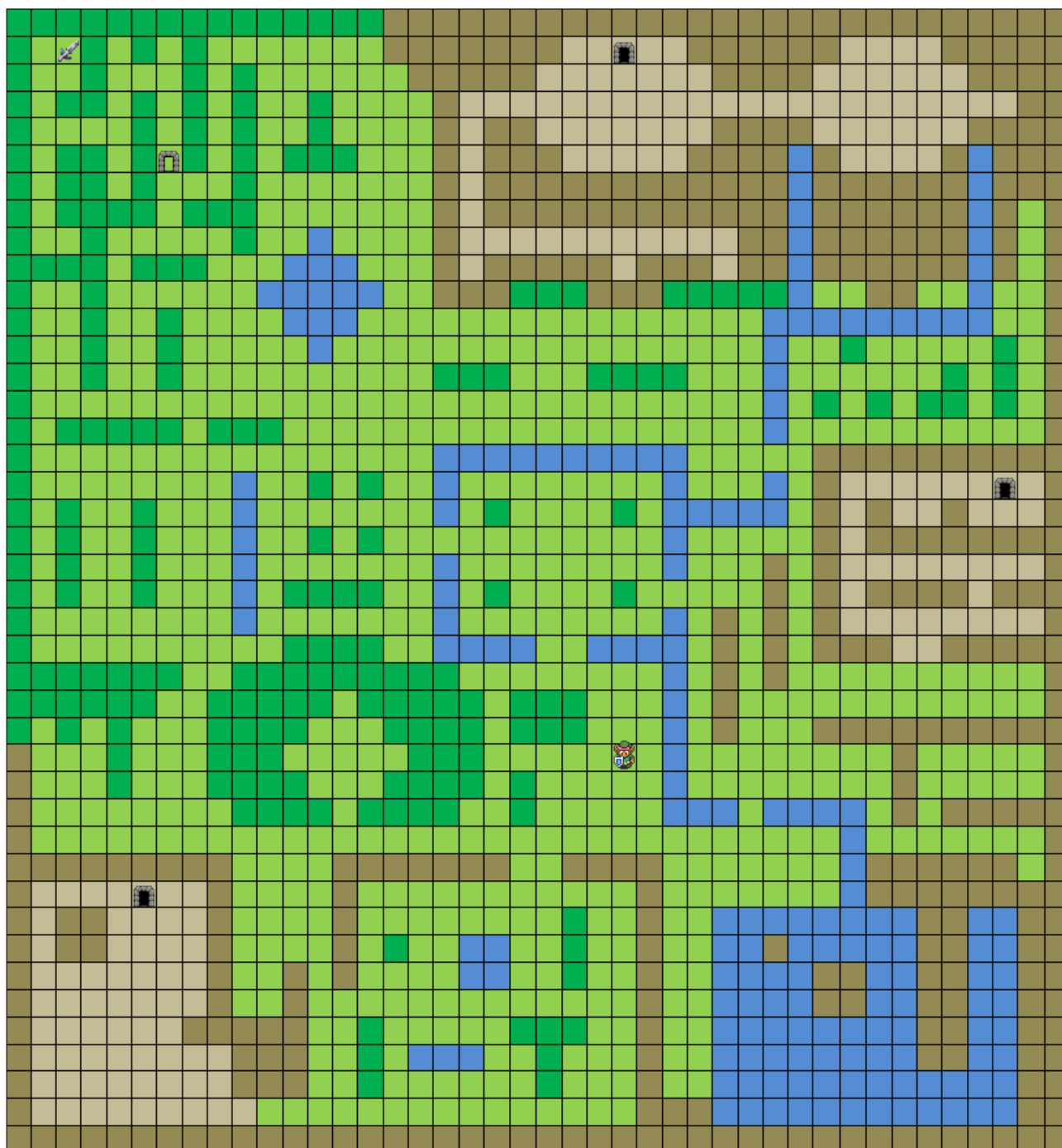


Figura 3 - Mapa do reino de Hyrule

Os **três pingentes da virtude** estão localizados dentro de **Dungeons**, as quais estão identificadas no mapa pelos portões de entrada. O mapa de cada Dungeon é mostrado na Figura 4, onde o portão marca o ponto de entrada/saída e o pingente identifica a posição do pingente da virtude.

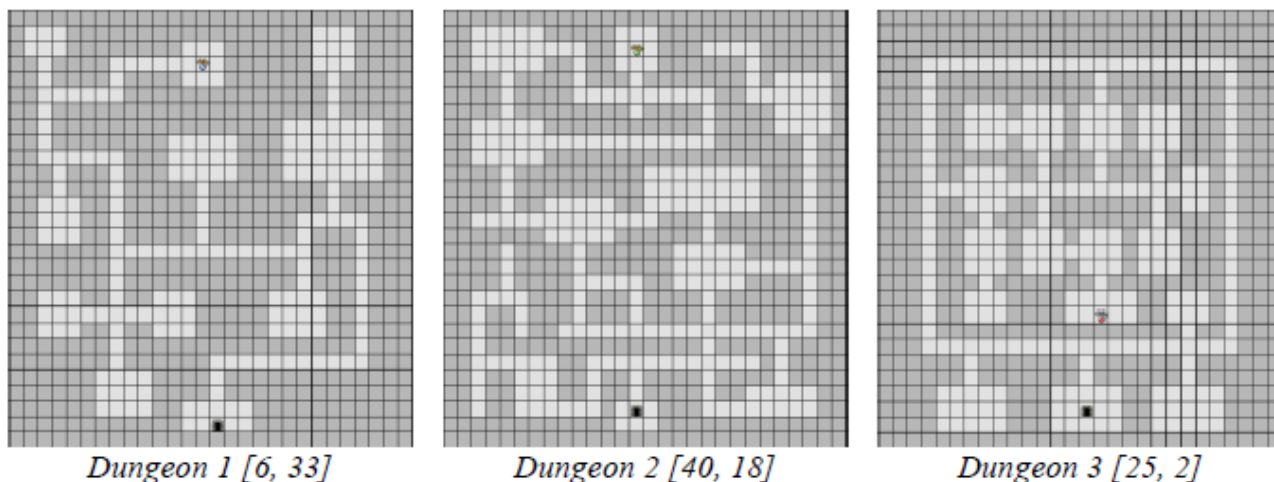


Figura 4 – Mapa das Dungeons do reino de Hyrule.

Dentro das Dungeons, somente é possível caminhar pelas regiões mais claras identificadas no mapa. O custo para andar nesse tipo de terreno é de +10.

Link inicia sua jornada na posição [25, 28] e termina após reunir os três pingentes da virtude e chegar até a entrada de **Lost Woods** (posição [7, 6]), onde ele poderá encontrar a **Master Sword**.

Informações Adicionais:

- O mapa principal deve ser representado por uma matriz 42 x 42 (igual à mostrada na Figura 3). As Dungeons também devem ser representadas por matrizes de tamanho 28 x 28 (iguais às mostradas na Figura 4).
- O agente sempre **inicia** a jornada na casa do Link (ponto onde está o Link no mapa [25, 28]).
- O agente sempre **termina** a sua jornada ao chegar à entrada de **Lost Woods**.
- Para encontrar a melhor ordem para pegar os pingentes é necessário resolver o problema do **caixeiro viajante (travelling salesman)**. Para isto considere que antes do agente chegar à entrada de **Lost Woods** ele deverá iniciar a jornada na casa de Link e voltar a esta casa com todos os pingentes coletados. Resolva tal problema usando alguma das técnicas de busca especificadas na disciplina.
- Ao entrar em uma **Dungeon**, o agente deve encontrar o melhor caminho até o pingente e depois retornar a entrada para sair da Dungeon e retornar para o mapa principal.
- Os pingentes podem ser **coletados em qualquer ordem**.
- O agente não pode andar na diagonal, somente na **vertical** e na **horizontal**.
- Deve existir uma maneira de **visualizar os movimentos** do agente, mesmo que a interface seja bem simples. Podendo até mesmo ser uma matriz desenhada e atualizada no console.
- **Os mapas devem ser configuráveis**, ou seja, deve ser possível modificar o tipo de terreno em cada local. O mapa pode ser lido de um arquivo de texto ou deve ser facilmente editável no código.
- O programa deve **exibir o custo do caminho** percorrido pelo agente enquanto ele se movimenta pelo mapa e também o **custo final** ao terminar a execução.
- O programa pode ser implementado em **qualquer linguagem de programação**.
- O trabalho pode ser feito **individualmente** ou em **grupos** de no máximo 3 pessoas.
- O programa deve ser apresentado por **todos os membros do grupo**:

- O membro do grupo que **não comparecer** receberá nota **zero**;
- O membro do grupo que **não souber explicar** algo relacionado ao trabalho perderá a metade dos pontos.

Dica: para resolver o problema de busca neste trabalho deve-se dividir o processo em pequenas etapas. Inicialmente realiza-se uma busca para encontrar o melhor caminho para chegar à primeira *Dungeon*. Ao entrar na *Dungeon* realiza-se uma nova busca para encontrar o melhor caminho dentro da *Dungeon* para chegar até o Pingente. Ao sair da *Dungeon*, busca-se o melhor caminho até a próxima *Dungeon* e repete-se o processo até chegar ao destino final;

Forma de Avaliação:

Será avaliado se:

- (1) O trabalho atendeu a todos os requisitos especificados anteriormente;
- (2) Os algoritmos foram implementados e aplicados de forma correta;
- (3) O código foi devidamente organizado;
- (4) O trabalho foi apresentado corretamente.

Bônus:

A interface gráfica não é o objetivo desse trabalho, mas quem implementar uma “**boa**” **interface gráfica** (2D ou 3D) para representar o ambiente e o agente receberá até 2 pontos extras na nota.

Data de Entrega:

23/06/2022

Forma de Entrega:

Todos os arquivos do programa deverão ser compactados (“zipados”). Faça um upload do arquivo compactado por meio do link disponível na página da disciplina, no moodle. Observa-se que dentre os arquivos compactados deverá existir um arquivo executável que permitirá visualizar o comportamento do código bem como um arquivo com instruções para uso do executável (pequeno manual do sistema).

Trabalhos entregues com atraso perderão 0.5 pontos para cada dia de atraso