



Universidade Estadual do Piauí - UESPI  
Centro de Tecnologia e Urbanismo - CTU  
Bacharelado em Ciência da Computação  
Disciplina: Inteligência Artificial  
Professora: Lianna Mara Castro Duarte

Obs.: Trabalho inspirado em estudo de problemática do prof. Edirlei Soares de Lima da Universidade Europeia – IADE de Lisbon, Portugal

### Trabalho Cavaleiros do Zodíaco – Busca Heurística (A\*)

**Descrição:** Durante o torneio da Guerra Galáctica, os **Cavaleiros de Bronze** descobrem que “Saori é a reencarnação de Atena e que o **Grande Mestre** tentou matá-la ainda bebê. Decididos a apoiar Saori, os Cavaleiros de Bronze partem para o **Santuário** para enfrentar o Grande Mestre. Ao chegar ao Santuário, Saori e os Cavaleiros são recepcionados por Tremy, um Cavaleiro de Prata, que ataca o grupo e atinge Saori com uma flecha mortal. Para salvar Atena, os Cavaleiros devem percorrer um caminho composto **pelas 12 Casas do Zodíaco**, cada uma protegida por um **Cavaleiro de Ouro**, e chegar à casa do Grande Mestre, o único capaz de remover a flecha do peito de Saori. Para complicar ainda mais, os Cavaleiros têm um **prazo máximo de 12 horas** para realizar essa tarefa!

O seu objetivo é ajudar **Seiya, Shiryu, Hyoga, Shun e Ikki** a passar pelas 12 Casas do Zodíaco, derrotando todos os Cavaleiros de Ouro e salvando Atena o mais rápido possível!”.



Figura 1. Os Cavaleiros de Bronze.



Figura 2. As 12 Casas do Zodíaco.

O Trabalho consiste em implementar um agente capaz de guiar autonomamente Seiya, Shiryu, Hyoga, Shun e Ikki pelas 12 Casas do Zodíaco, planejando a melhor forma de derrotar os 12 Cavaleiros de Ouro e salvar Atena. Para isso, você deve utilizar o algoritmo de busca heurística A\*.

O agente deve ser capaz de calcular automaticamente a melhor rota para percorrer as 12 Casas do Zodíaco e derrotar os 12 Cavaleiros de Ouro no menor tempo possível.

O mapa das 12 casas do zodíaco é mostrado na Figura 3.

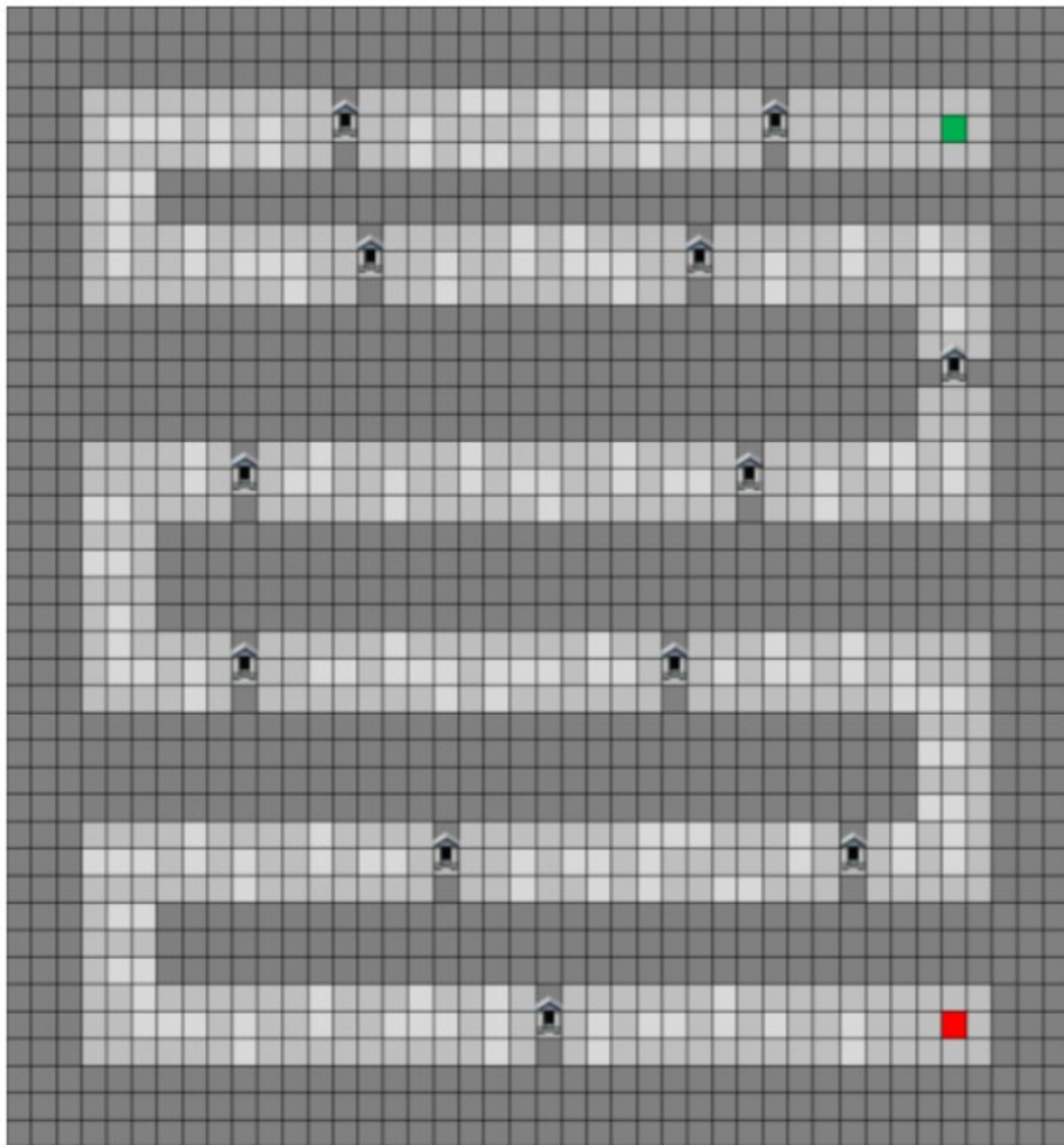


Figura 3. Mapa das 12 Casas do Zodíaco.

No caminho das 12 Casas do Zodíaco existem 3 tipos de terrenos: montanhoso (região cinza escuro), plano (região cinza) e rochoso (região cinza claro). Para passar por cada tipo de terreno, os Cavaleiros gastam uma determinada quantidade de tempo:

- Montanhoso: +200 minutos
- Plano: +1 minuto
- Rochoso: +5 minutos

Os Cavaleiros de Bronze iniciam a sua jornada na entrada do santuário (região em vermelho no mapa) e terminam ao chegar à casa do Grande Mestre (região verde no

mapa).

Ao chegar a uma Casa do Zodíaco, o agente deve decidir quais Cavaleiros vão lutar contra o Cavaleiro de Ouro que protege a casa. Cada Cavaleiro de Ouro apresenta um nível de dificuldade diferente. Este nível determina o tempo gasto pelos Cavaleiros de Bronze para poder vencê-lo e avançar para a próxima Casa.

A Tabela 1 mostra os níveis de dificuldade das 12 Casas do Zodíaco.

	<b>Casa</b>	<b>Dificuldade</b>
1°	Casa de Áries	50
2°	Casa de Touro	55
3°	Casa de Gêmeos	60
4°	Casa de Câncer	70
5°	Casa de Leão	75
6°	Casa de Virgem	80
7°	Casa de Libra	85
8°	Casa de Escorpião	90
9°	Casa de Sagitário	95
10°	Casa de Capricórnio	100
11°	Casa de Aquário	110
12°	Casa de Peixes	120

Tabela 1. Níveis de dificuldade das 12 Casas do Zodíaco.

O número de Cavaleiros de Bronze participando das batalhas contra os Cavaleiros de Ouro influencia o tempo gasto na batalha. Além disso, cada Cavaleiro possui um determinado nível de poder cósmico que também influencia no tempo gasto nas batalhas. Quanto mais cavaleiros lutando, mais rápido o Cavaleiro de Ouro será derrotado.

A Tabela 2 mostra o poder cósmico dos Cavaleiros de Bronze.

<b>Cavaleiro</b>	<b>Poder Cósmico</b>
Seiya	1.5
Shiryu	1.4
Hyoga	1.3
Shun	1.2
Ikki	1.1

Tabela 2. Poder cósmico dos Cavaleiros de Bronze.

O tempo gasto nas batalhas contra os Cavaleiros de Ouro é dado pela função:

$$Tempo = \frac{\sum Dificuldade da Casa}{Poder Cosmico dos Cavaleiros Participando da Batalha}$$

Além do poder cósmico cada Cavaleiro de Bronze também possui **5 pontos de energia**. Ao participar de uma batalha, o cavaleiro perde -1 ponto de energia. Se o Cavaleiro perder todos os pontos de energia ele morre.

Informações adicionais:

- O mapa principal deve ser representado por uma matriz 42 x 42 (igual à mostrada na Figura 3).
- O agente sempre inicia a jornada na entrada do santuário (região em vermelho no mapa).
- O agente não pode andar na diagonal, somente na vertical e na horizontal.
- O agente obrigatoriamente deve utilizar um algoritmo de busca para encontrar o melhor caminho e planejar as batalhas.
- 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.
- A dificuldade das casas e o poder cósmico dos Cavaleiros de Bronze devem ser configuráveis e facilmente editáveis.
- O programa deve exibir o custo do caminho percorrido pelo agente enquanto ele se movimenta pelo mapa e o custo final ao terminar a execução.
- O programa pode ser implementado em qualquer linguagem.
- O trabalho pode ser feito individualmente ou em grupos de no máximo 2 pessoas.

Dicas:

- Neste trabalho existem dois problemas distintos:
  1. Encontrar o melhor caminho para passar pelas 12 Casa do Zodíaco e chegar até a Casa do Grande Mestre;
  2. Encontrar a melhor ordem de equipes para lutar contra os Cavaleiros de Ouro.
- Os dois problemas podem ser resolvidos individualmente ou tratando ambos em um único problema. Você deve definir a melhor maneira de estruturar a sua solução, use sua criatividade.

### Considerações:

- 1) O trabalho pode ser desenvolvido em duplas. Em hipótese nenhuma serão aceitos grupos com mais de dois alunos.
- 2) Além da implementação do algoritmo de busca, o programa deverá conter um módulo de interface com o usuário, para exibir uma animação correspondente à **solução encontrada**. Existem boas bibliotecas disponíveis para a maioria das linguagens, como a PyGame (Python), Allegro (C/C++), JMonkeyEngine (Java), Unity (C#), dentre outros.
- 3) O código-fonte que não puder ser compilado/interpretado diretamente não será avaliado, resultando em nota 0 para o(s) alunos(s).
- 4) Qualquer tipo de cópia (da internet, de outros alunos, etc.) resultará em nota 0.
- 5) Os alunos deverão gravar um vídeo curto apresentando o trabalho (se feito em dupla ambos devem participar) explicando código e execução do programa desenvolvido. Ambos serão avaliados individualmente neste quesito.

- 6) O código deve ser anexado, em conjunto com o vídeo explicativo no google Classroom. no item disponível para postagem do trabalho por um dos membros da dupla no prazo definido.

## **AVALIAÇÃO**

<b>ITEM</b>	<b>PONTUAÇÃO</b>	<b>PONTUAÇÃO OBTIDA</b>
Implementação todos os requisitos especificados anteriormente	3,5	
Animação	1,0	
Código-fonte: Modularização	1,0	
Código-fonte: Legibilidade	1,0	
Código-fonte: Indentação	0,5	
Avaliação Individual (explicação)	2,0	
Criatividade: neste item, será avaliado a capacidade de criação dos alunos além do que foi especificado no trabalho.	1,0	