

# INF1771 - INTELIGÊNCIA ARTIFICIAL

## TRABALHO 1 – BUSCA HEURÍSTICA

### Descrição:

“A **Barbie** é a garota mais linda e popular do **Mundo da Barbie!** Porém, o que poucos sabem, é que a Barbie também é uma excelente **programadora!** Desde criança, Barbie sempre sonhou em ser a “**melhor programadora do mundo**”!

Para obter o título de “**melhor programadora do mundo**”, Barbie deve ganhar o **1º Concurso Mundial de Desenvolvimento de Softwares**. Porém, para poder participar do concurso, a Barbie precisa reunir uma **equipe de programadores**.

Ao se deparar com este problema, Barbie rapidamente lembrou-se dos seus amigos que estavam aprendendo a programar: **Suzy, Polly, Mary, Carly, Ken e Brandon**. Para poder participar do concurso, Barbie precisa **convencer três amigos a fazerem parte da equipe de programadores!**

O seu objetivo é encontrar os amigos da Barbie e tentar convence-los a entrar para a equipe de programadores! Durante a jornada, lembre-se que a **Barbie nunca pode perder o seu glamour!** Por isso, sempre que necessário refaça a **maquiagem com o Kit de Maquiagem da Barbie!**”.



Figura 1. Barbie Programadora.



Figura 2. Kit de Maquiagem da Barbie.

O Trabalho 1 consiste em implementar um agente capaz de locomover-se autonomamente pelo Mundo Barbie, explorar os diversos ambientes e convencer três amigos a participar do concursos mundial de programadores. Para isso, você deve utilizar o **algoritmo de busca heurística A\***.

O agente deve ser capaz de calcular automaticamente a **melhor rota para encontrar e convencer três amigos a participar do concurso, e retornar até a Casa da Barbie**.

O mapa do **Mundo da Barbie** é mostrado na Figura 3.

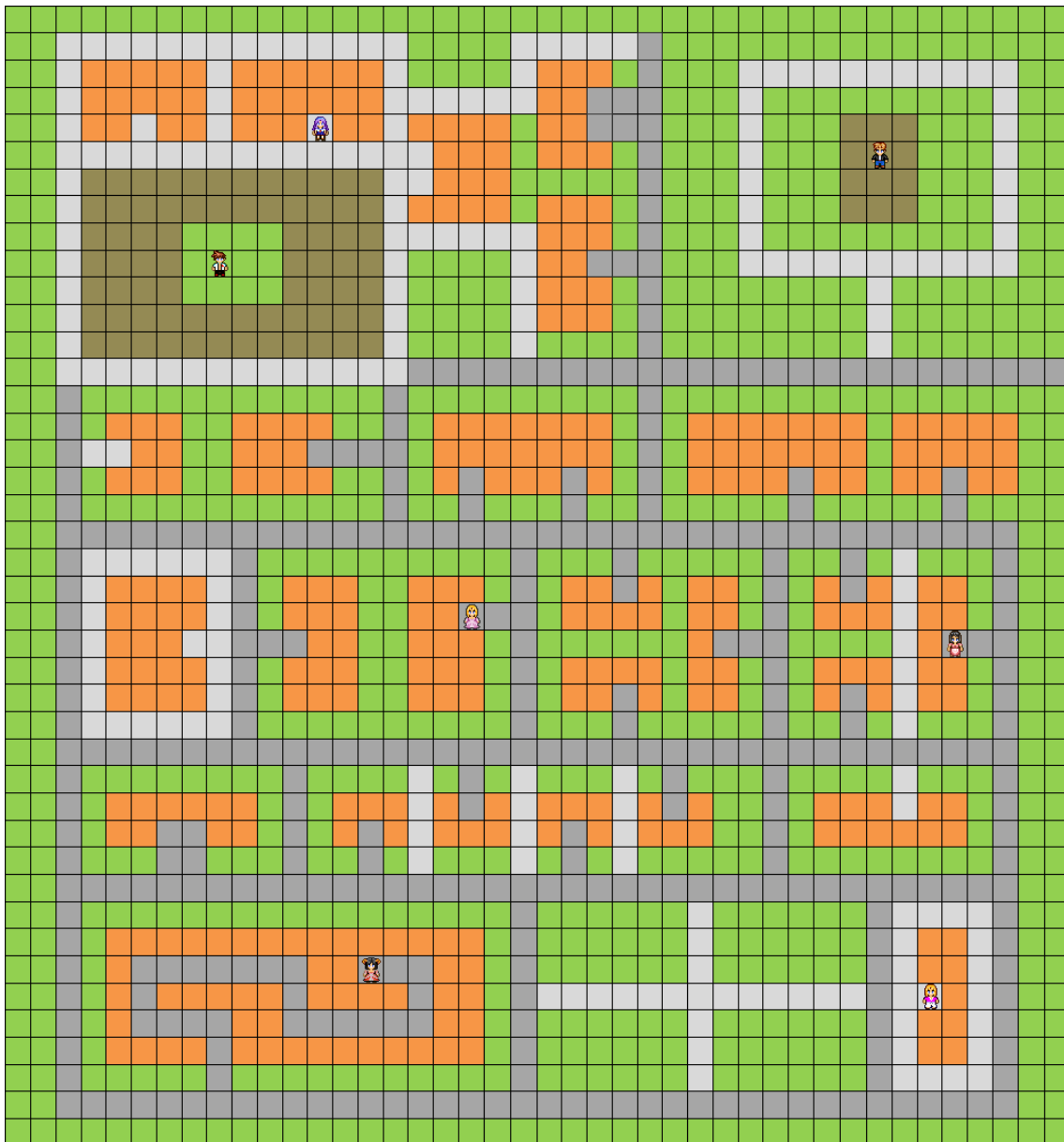


Figura 3. Mapa do Mundo da Barbie.

O Mundo da Barbie é formado por 5 **tipos de terrenos**: asfalto (região cinza escuro), grama (região verde), terra (região marrom), paralelepípedo (região cinza claro) e edifícios (região laranja).

A Barbie nunca pode perder a classe, então mesmo nesta aventura ela está utilizando o seu **salto agulha**. Dessa forma, cada tipo de terreno exige uma determinada quantidade de esforço, o que faz com a Barbie precise **refazer a sua maquiagem constantemente para continuar diva**.

**A quantidade de maquiagem gasta para passar por cada tipo de terreno são os seguintes:**

- **Asfalto** – Custo: +1
- **Terra** – Custo: +3
- **Gramma** – Custo: +5
- **Paralelepípedo** – Custo: +10

A Barbie **nunca pode passar por regiões de edifícios** (regiões de cor laranja no mapa da Figura 3).

As localizações dos 6 melhores amigos da Barbie estão definidas na Figura 3. Ao encontrar um amigo, a Barbie deve tentar convence-lo a participar do concurso, porém **o amigo pode aceitar ou recusa o convite**. Se o convite recusado, a Barbie deve tentar encontrar outro amigo.

A Barbie inicia sua jornada na **Casa da Barbie** (posição [19, 23] no mapa) e termina após ela convencer três amigos a participarem do concurso e retornar até a sua casa. A melhor rota para cumprir essa missão é a rota de menor custo levando em consideração o terreno.

#### **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 Casa da Barbie (ponto onde está a Barbie está no mapa [19, 23]).
- O agente sempre **termina** a sua jornada ao convencer três amigos e retornar até a Casa da Barbie (posição [19, 23]). O desenvolvimento do software para o concurso será uma aventura para o Trabalho 2.
- O agente não pode andar na diagonal, somente na **vertical** e na **horizontal**.
- Os amigos podem ser **convencidos em qualquer ordem**. Porém, ordens diferentes vão resultaram em custos totais diferentes. Além disso, este é um **problema não-determinístico**, onde não é possível prever se o amigo vai aceitar o convite para participar do concurso.
- Devem existir **somente três amigos que vão aceitar o convite da Barbie**. As indicações de quais amigos vão aceitar o convite da Barbie deve ser **sorteada na inicialização do programa**, porém o algoritmo de busca não pode ter acesso a essa informação durante o processo de busca.

- 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**.
- O trabalho pode ser feito **individualmente** ou em **grupos** de no máximo 3 pessoas.
- O programa deve ser apresentado durante a aula 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 5.0 pontos.

#### **Dicas:**

- Como o agente não sabe quais amigos vão aceitar os convites, assuma que no pior dos casos será necessário visitar todos os amigos.
- Note que este problema é semelhante ao problema do Caixeiro Viajante (Travelling Salesman Problem). É necessário encontrar a melhor rota para visitar todos os amigos uma vez. No trabalho não é obrigatório a resolução deste problema, mas é única maneira de garantir o melhor custo.
- Implemente a função de busca de uma forma genérica, pois será necessário executa-la múltiplas vezes para diferentes destinos.

#### **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 em sala de aula;

**Bônus:**

- (1) 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.
- (2) O programa que conseguir resolver o problema proposto com o menor custo, receberá 2 pontos extras na nota. Em caso de empate, o critério de desempate será a velocidade de execução do algoritmo de busca. Nesse caso, o trabalho deverá ter implementado um mecanismo para calcular o tempo gasto pelo algoritmo. Para poder participar da competição, o trabalho deverá permitir que os amigos que vão aceitar os convites da Barbie sejam definidos manualmente.

**Data de Entrega:**

01/10

**Forma de Entrega:**

O programa deve ser apresentado na aula do dia 01/10 (segunda) e enviando até o mesmo dia para o email [edirlei.slima@gmail.com](mailto:edirlei.slima@gmail.com).

Trabalhos entregues atrasados perderam 0.5 pontos para cada dia de atraso.