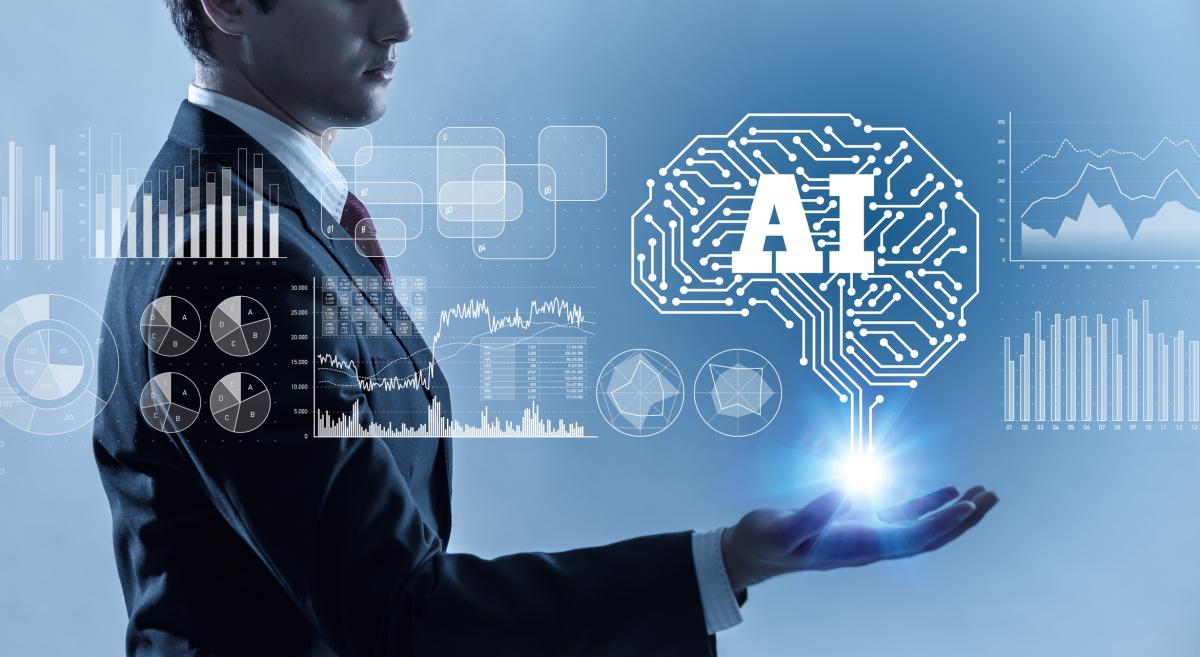


Licenciatura Engenharia

Informática & Telecomunicação



**Docente:** Estanislau Lima

**Discente:** Fredson Bandeira

HW-lecture 2 and 3

**1 Towers of Hanoi**

1. Visto que cada disco só pode ficar ordenado nos pinos se o maior tiver em baixo e o mais pequeno tiver em cima. Então podemos dizer que os discos N são numerados por ordem crescente de tamanho 1 á n. cada pino como uma lista ordenada de números inteiros correspondentes a que os discos estão nessa cavilha. As três listas, correspondentes a que discos estão em que cavilha.
2. 4
3. ([1, ..., n], [], [])
4. Se colocarmos o primeiro inteiro de qualquer lista e empurrá-lo para a frente de outra lista, desde que é menor do que o número inteiro atualmente na frente da lista que está a cavilha a ser movida para.
5. O estado é o mesmo que ([], [], [1, ..., n])

**2 Search and Heuristics**

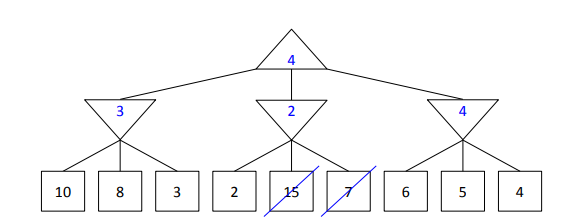
1. O tamanho do espaço de estado é de 4MN (Vmax + 1). A representação do estado é (direção virada, x, y, velocidade). Nota que a velocidade pode assumir qualquer valor em {0, ..., Vmax}.
2. Não, a distância de Manhattan não é uma heurística admissível. O agente pode mover-se a uma velocidade média de mais de 1 (primeiro acelerando até Vmax e depois abrandando até 0 à medida que alcança o objetivo), e assim pode alcançar o objetivo em menos etapas de tempo do que há quadrados entre ela e a meta. Um exemplo específico: o alvo é 6 quadrados e a velocidade do agente já é de 4. Ao tomar apenas 4 ações lentas, alcança o objetivo com uma velocidade 0.
3. Aqui estão algumas, por ordem dos mais fracos aos mais fortes:

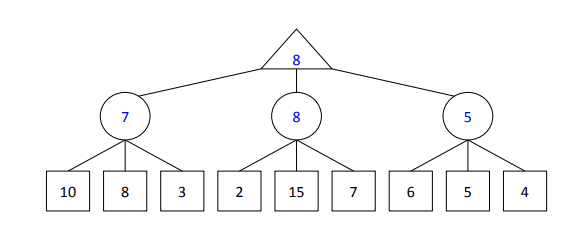
O número de voltas necessárias para o agente enfrentar o objetivo.

Considerar um relaxamento do problema onde não há paredes, o agente pode virar e mudar de velocidade arbitrariamente. Neste problema descontraído, o agente movia-se com Vmax, e de repente parava no objetivo, demorando assim dmanhattan/Vmax.

1. Se a função heurística estiver limitada, então a pesquisa gráfica A\* acabaria por visitar todos os nós, e encontraria um caminho para o estado de objetivo, se é que existe um. Uma heurística inadmissível não garante a otimização como pode fazer com que o bom objetivo ideal pareça muito distante, e levá-lo a um objetivo subotimizado.
2. A heurística admissível não garante necessariamente uma otimização; apenas lhes é garantido o retorno de um ótimo solução, se forem também consistentes.
3. O tempo para resolver um problema de pesquisa A\* é função de dois factores: o número de nós expandidos, e o tempo gasto por nó. Um heurístico inadmissível pode ser mais rápido de calcular, levando a uma solução que é obtida mais rápido devido a menos tempo gasto por nó. Pode também ser uma estimativa mais próxima da função de custo real (ainda que às vezes sobrestimará!), expandindo assim menos nós. Perdemos a garantia de otimização ao utilizar um heurística inadmissível. Mas por vezes podemos não ter problemas em encontrar uma solução sub-óptima para um problema de busca.

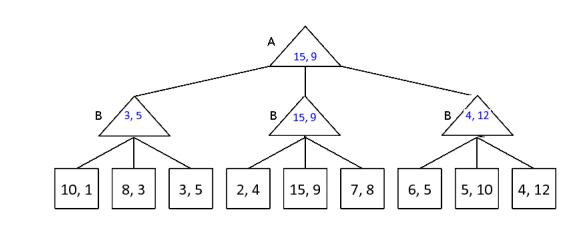
**Games**

1. 



d) Nenhum nó pode ser podado. Haverá sempre a possibilidade de alguns folha mais abaixo o ramo terá um valor muito elevado, o que aumenta o valor médio global.

**Nonzero-sum Games**

1)

2) Nenhum nó pode ser podado. Porque este jogo não é zero, pode existir um nó de folha em qualquer parte da árvore que seja bom tanto para o jogador A como para o jogador B.