

# Sistemas Evolutivos e Aplicados à Robótica

- Ivan Sousa
- Leandro Bellini
- Pedro Fini

# Objetivo

- Aplicar conceitos de A.G. em um sistema básico para determinação de melhores rotas em veículos autônomos

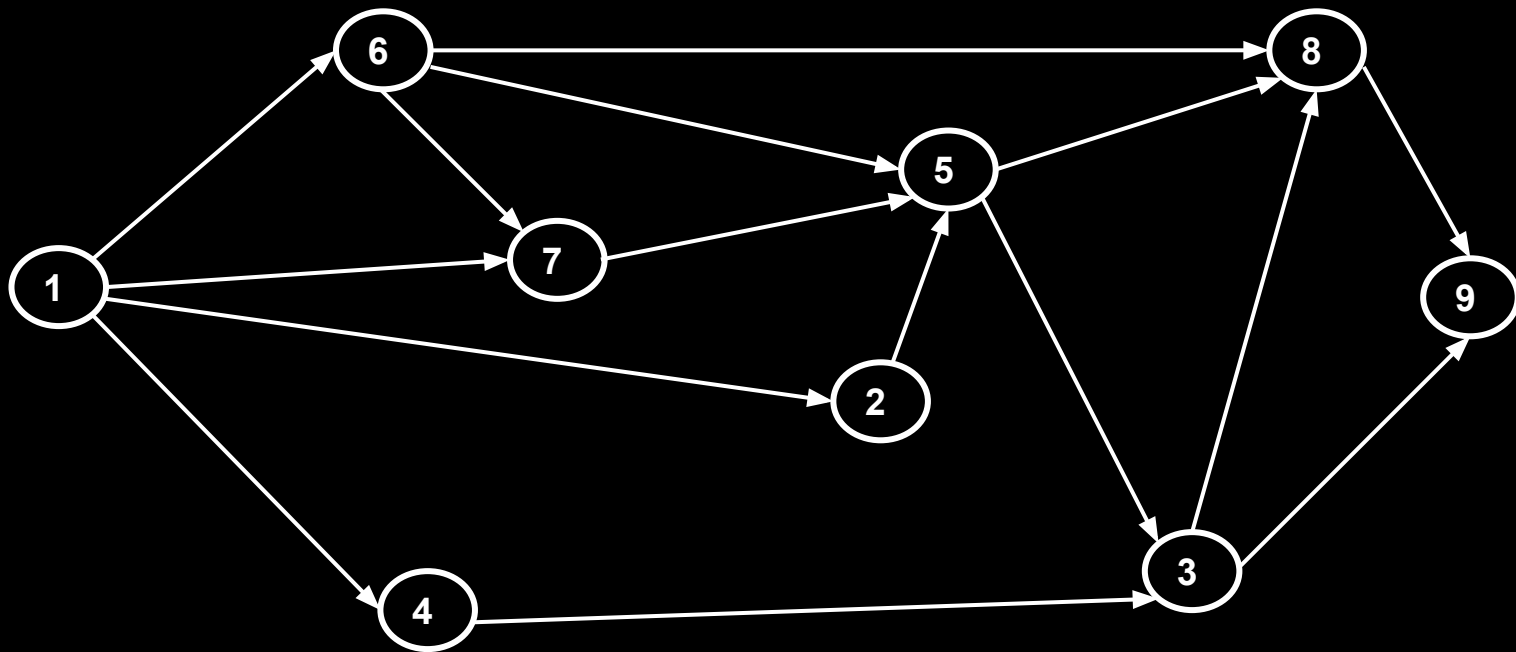
# Execução do Projeto

- Determinação de parâmetros;
- Caminho mínimo para cada parâmetro isolado;
- *Crossover* para as soluções encontradas;
- Mutação dos caminhos existentes;
- Análise do *fitness* obtido.

# Grafo

- Aplicação de grafos em caminho
  - Nó: Cidade
  - Arestas: Caminhos

# Grafo



# Algoritmo de Dijkstra

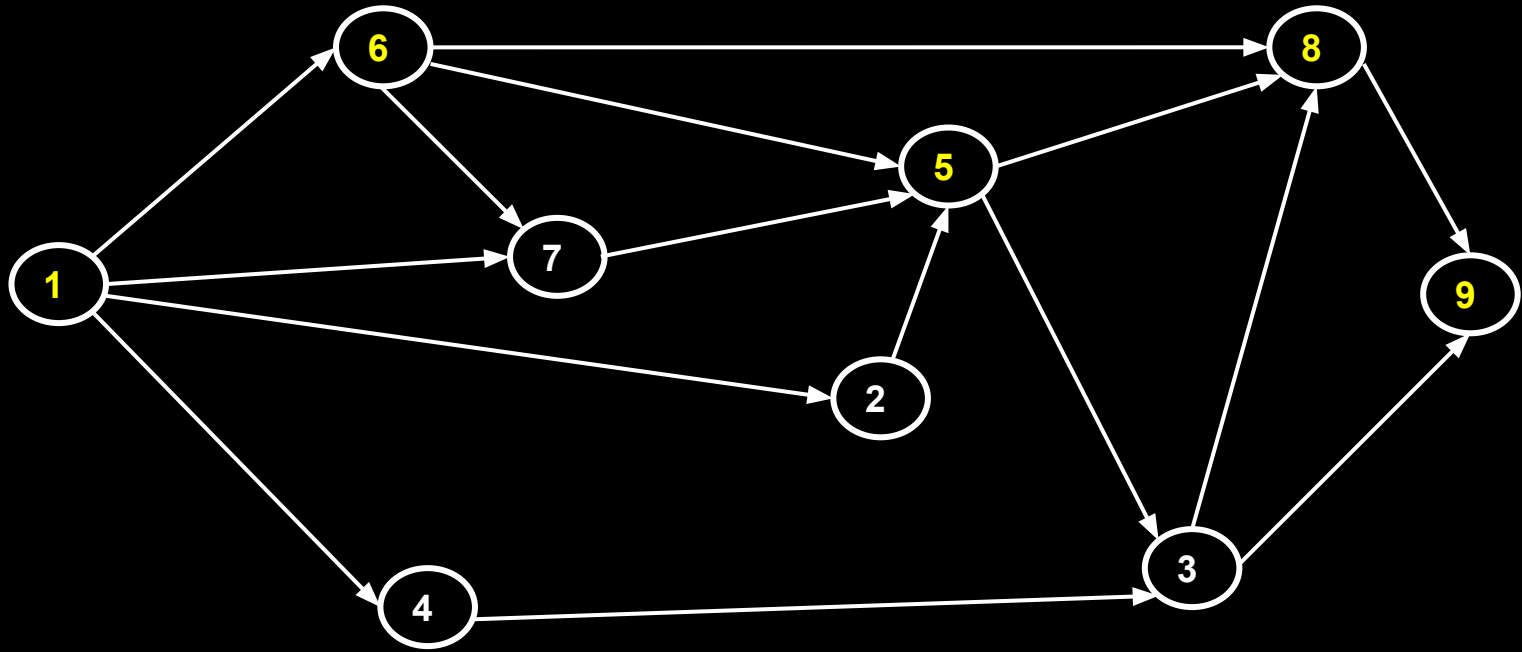
Melhores caminhos conforme a necessidade

Tempo

Custo

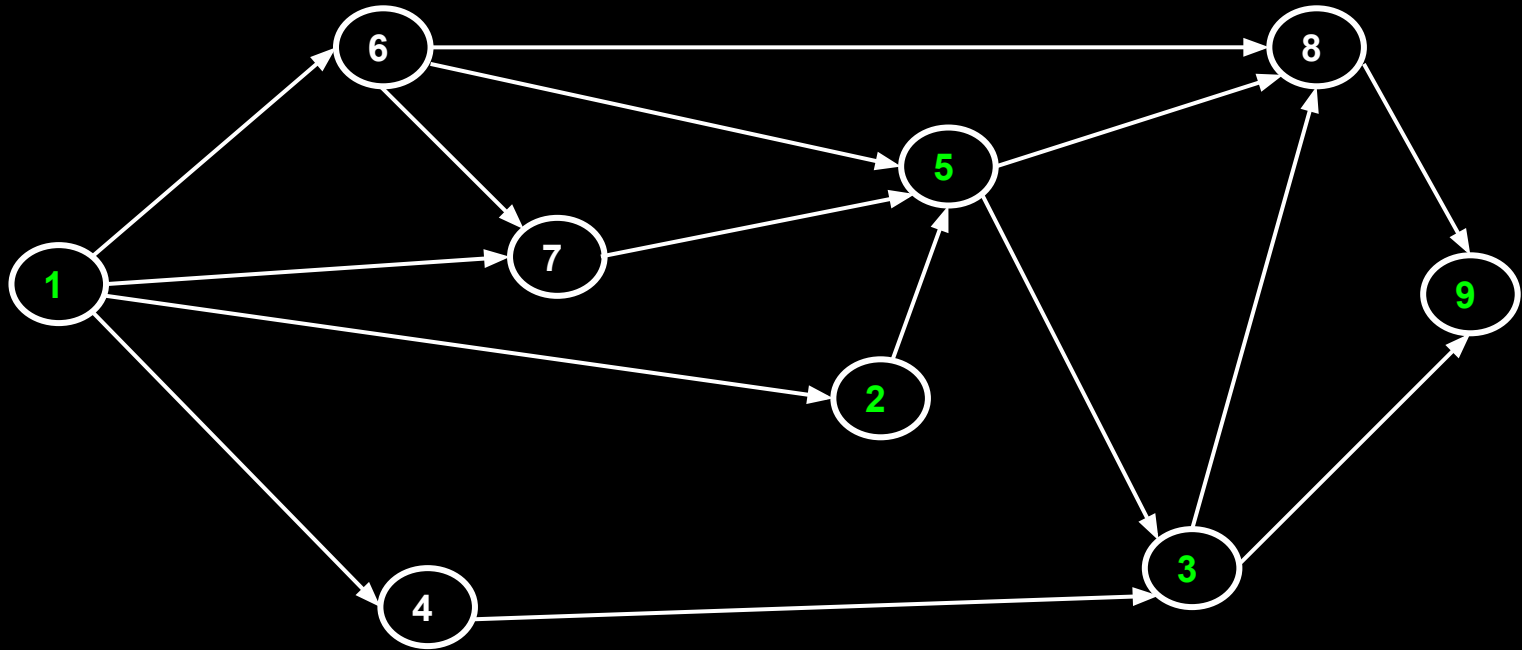
Segurança

# Melhor Caminho em Tempo



{Pedágio, Perigo, Tempo}

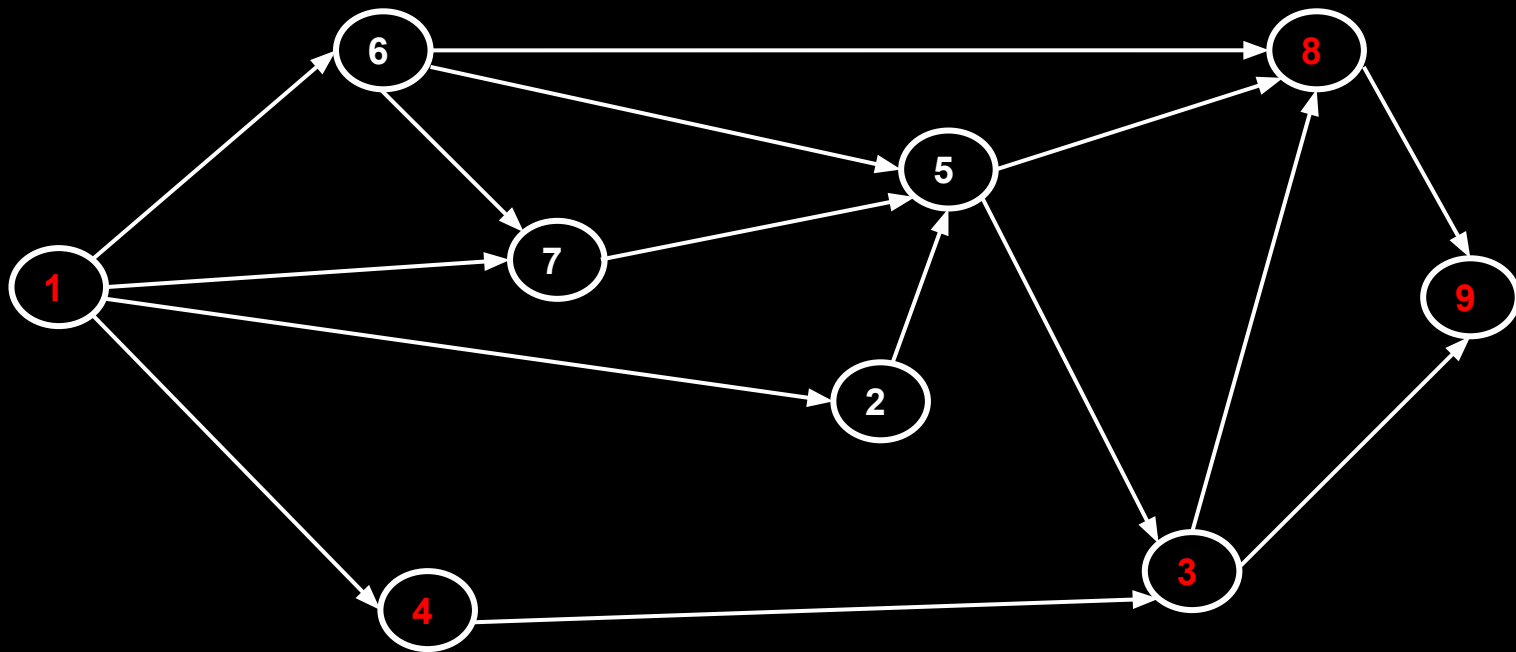
# Melhor Caminho em Custo



{Pedágio, Perigo, Tempo}



# Melhor Caminho em Segurança



{Pedágio, Perigo, Tempo}

# População Inicial

- População inicial obtida através do resultado do Dijkstra

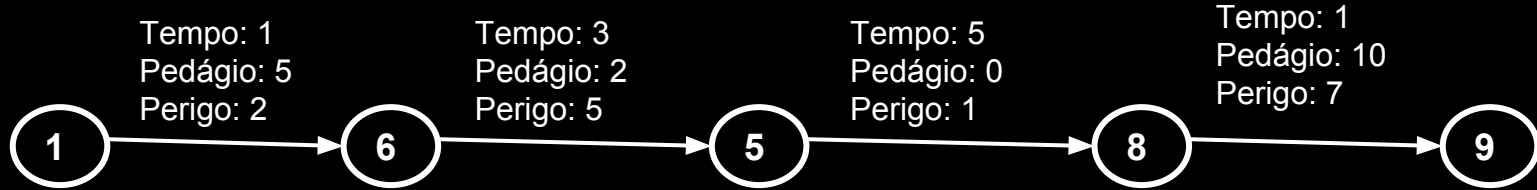
Melhor caminho para tempo: (1) -> (6) -> (5) -> (8) -> (9)

Melhor caminho para custo: (1) -> (2) -> (5) -> (3) -> (9)

Melhor caminho para segurança: (1) -> (4) -> (3) -> (8) -> (9)

# Fitness

- Soma dos pesos das arestas do caminho



$$\text{Fitness} = \frac{1}{\sum p_{\text{Aresta}}} \times 1000 = 23,8$$

# Progresso Atual

- Aplicação de Dijkstra;
- Determinação de caminhos iniciais;
- Cálculo do Fitness dos caminhos (Soma pesos);
- Seleção dos melhores caminhos (Pais);
- *Crossover* com objetivo de gerar caminhos ótimos

```
gcc -o dijkstra dijkstra.c  
./dijkstra
```

Caminho Tempo:

[1] [6] [5] [8] [9]

Tempo: 20, Perigo: 45, Pedagio: 33 ---> Fitness: 10.204082

Caminho Perigo:

[1] [4] [3] [8] [9]

Tempo: 35, Perigo: 20, Pedagio: 45 ---> Fitness: 10.000000

Caminho Pedagio:

[1] [2] [5] [3] [9]

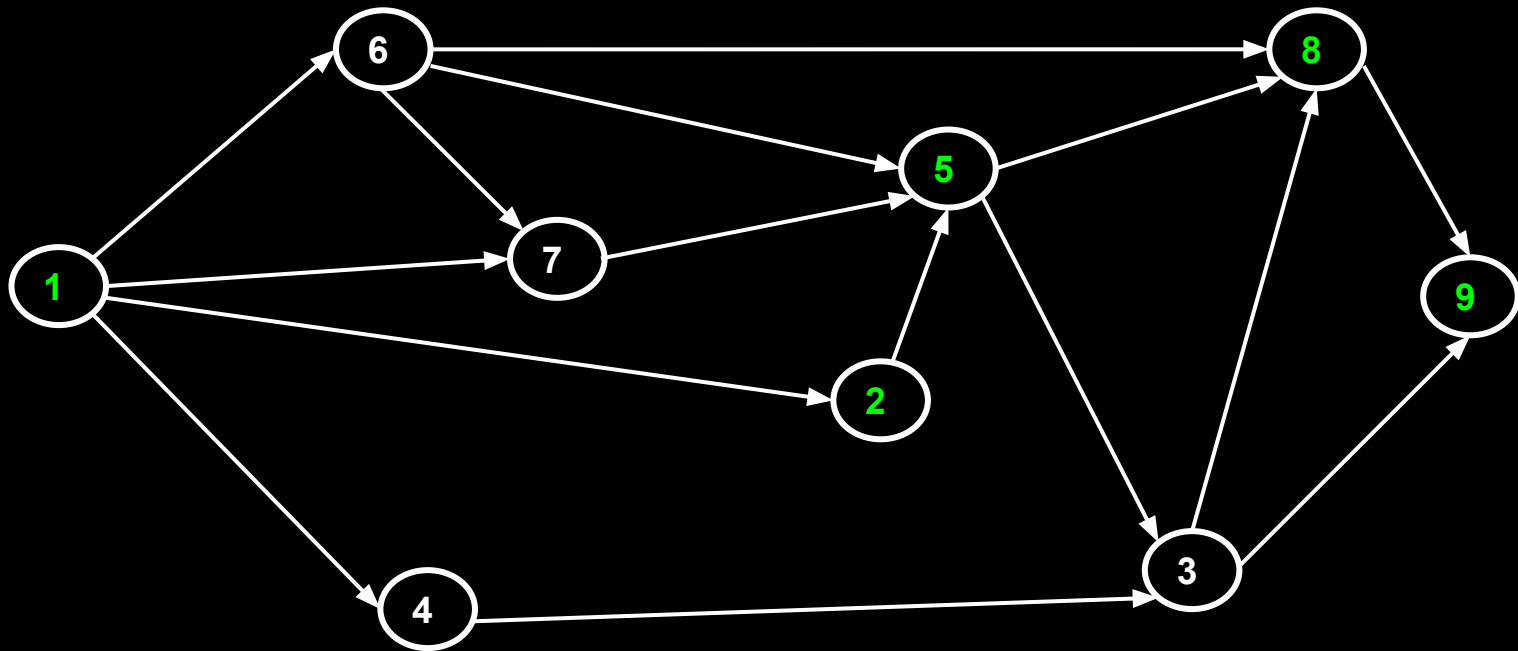
Tempo: 38, Perigo: 33, Pedagio: 25 ---> Fitness: 10.416667

Caminho Otimizado:

[1] [2] [5] [8] [9]

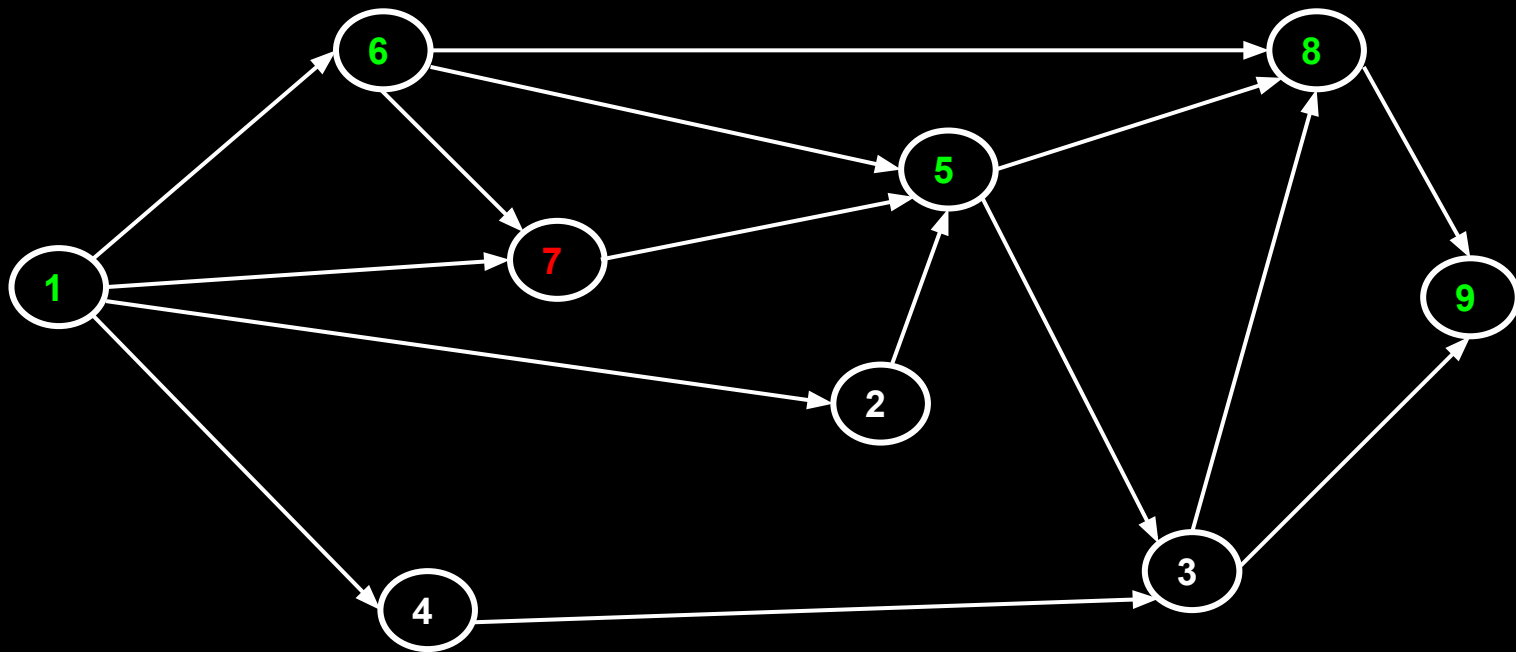
Fitness --> 10.638297

# Recombinação



{Pedágio, Perigo, Tempo}

# Mutação



A escolha do Nó para a mutação é aleatório, e verifica se existe arestas

# Mutação

- População inicial conforme o tempo obtida através do resultado do Dijkstra

Melhor caminho para tempo: (1) -> (6) -> (5) -> (8) -> (9)

Melhor caminho para tempo: (1) -> (6) -> (7) -> (5) -> (8) -> (9)

Fitness Antigo: 10,2

Fitness Novo: 7,9

Nesse caso não deu certo!



Obrigado!!!