

UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE INFORMÁTICA  
SISTEMAS INTELIGENTES  
2018.2



# Planejamento Turismo Multimodal em PDDL

EQUIPE:  
LUCAS BARROS DE ALMEIDA MACHADO (LBAM)  
RAFAEL MOTA ALVES (RMA7)

## Sumário

1.	<a href="#">Desafio</a> .....	2
2.	<a href="#">Implementação</a> .....	3
2.1.	<a href="#">Tipos e Objetos</a> .....	3
2.2.	<a href="#">Predicados</a> .....	3
2.3.	<a href="#">Ações</a> .....	4
2.4.	<a href="#">Restrições</a> .....	6
2.5.	<a href="#">Objetivo</a> .....	7
3.	<a href="#">Resultados</a> .....	8

# 1. Desafio

O desafio deste projeto consiste em implementar uma solução PDDL para o problema Turismo Multimodal descrito em [Turismo Multimodal 2018.2](#). Em suma, o problema limita-se em planejar um roteiro em que três turistas em uma cidade (João, Maria e José) possam visitar com sucesso os locais que desejam ir. Os turistas utilizam apenas bicicletas como meio de transporte (essas adquiridas por estações espalhadas pela cidade) e deve ser levado em conta algumas variáveis, como disponibilidade de bicicletas em cada estação e restrição de tempo do uso de uma bicicleta, além de algumas restrições, como João ter que visitar um local antes que outro e José e Maria terem que se encontrar para visitar um local. O mapa do problema se encontra abaixo, em que os pontos vermelhos identificam as estações de bicicleta e os pontos marrons identificam locais os quais os turistas podem visitar.

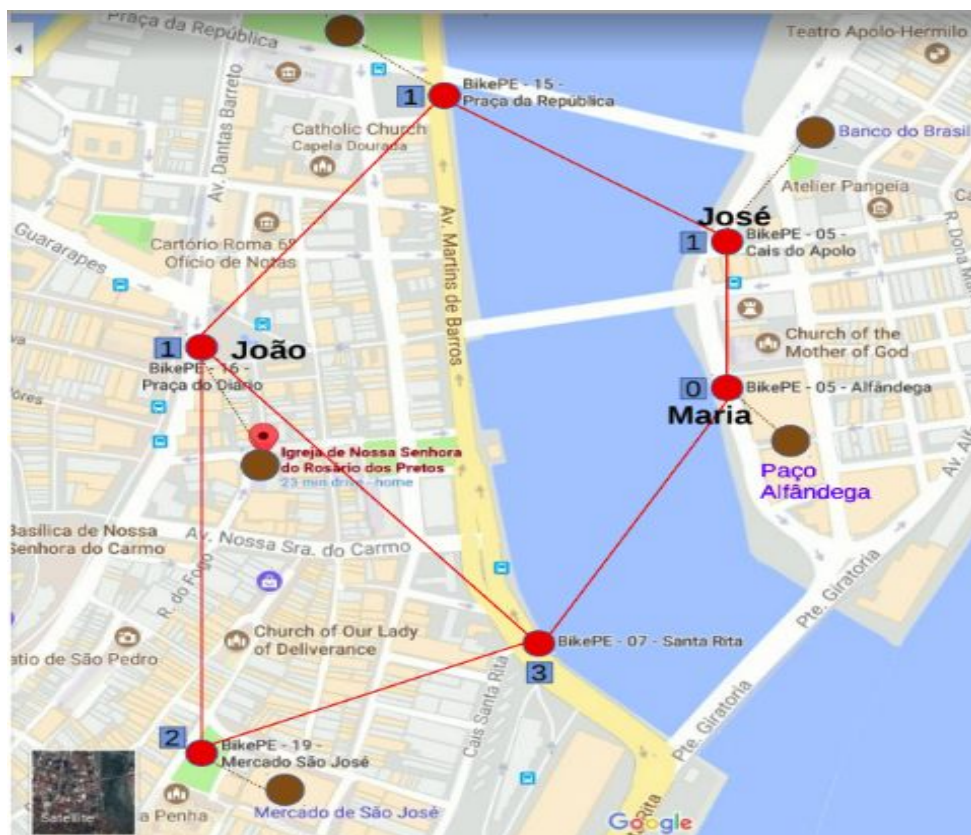


Imagem 1 - Mapa do problema Turismo Multimodal

## 2. Implementação

O projeto utiliza o requisito **strips**, que consiste no subconjunto básico do PDDL e também o **typing**, que traz o conceito de tipagem para o PDDL.

### 2.1 Tipos e Objetos

O tipos criados para resolver o problema são: **pessoa**, **bicicleta** e **lugar**

- **pessoa**: Representa as pessoas do problema: João, José e Maria, representados respectivamente pelos objetos: **joao**, **jose** e **maria**. Apenas os objetos desse tipo podem “executar” as ações do problema.
- **bicicleta**: Representa todas as bicicletas distribuídas pela cidade, que são representadas pelos objetos: **bike-1 a bike-8**, apenas a bicicletas podem ser movimentadas pelas pessoas.
- **lugar**: Esse tipo pode representar tanto atrações turísticas quanto pontos de coleta de bicicleta, apenas eles podem ser visitados e todas a movimentações são feitas de um para o outro. Os objetos: **igreja**, **praca-republica**, **mercado**, **paco-alfandega** e **banco-brasil** representam os pontos turísticos e os objetos: **bike-republica**, **bike-diario**, **bike-mercado**, **bike-santa-rita**, **bike-alfandega** e **bike-apolos** representam as estações de coleta de bicicletas.

### 2.2 Predicados

- **adjWalk**: O predicado adjWalk recebe como parâmetros **dois lugares**, e representa que a distância entre eles pode ser percorrida a pé, essa relação ocorre entre atrações turísticas e alguns pontos de coleta de bicicleta.
- **adjBike**: O predicado adjBike recebe como parâmetros **dois lugares**, e representa que a distância entre eles pode ser percorrida de bicicleta, respeitando a restrição dos 30 minutos, essa relação ocorre entre pontos de coleta de bicicleta.
- **personAt**: O predicado personAt recebe como parâmetros **uma pessoa** e **um lugar**, e indica se a pessoa está neste lugar.
- **bikeAt**: O predicado bikeAt recebe como parâmetros **uma bicicleta** e **um lugar**, e indica se a bicicleta está neste lugar.
- **hasBeen**: O predicado hasBeen recebe como parâmetros **uma pessoa** e **um lugar**, ele indica se a pessoa já esteve nesse lugar.
- **hasBike**: O predicado hasBike recebe como parâmetros **uma pessoa** e **uma bicicleta**, ele indica se a pessoa está com essa bicicleta no momento.
- **withBike**: O predicado withBike recebe como parâmetro **uma pessoa**, e indica se essa pessoa está com uma bicicleta no momento
- **canGetBike**: O predicado canGetBike recebe como parâmetro **uma pessoa**, e indica se essa pessoa pode pegar uma bicicleta no momento, respeitando a regra de que se deve esperar 5 minutos entre entregar uma bicicleta e pegar outra.

- **expiredTime:** O predicado expiredTime recebe como parâmetro **uma bicicleta**, e indica se o prazo de 30 minutos para entregar a bicicleta expirou ou vai expirar na próxima pedalada.
- **hasSubobjective:** O predicado hasSubobjective recebe como parâmetro **uma pessoa**, e indica se essa pessoa deve visitar algum lugar antes.
- **subobjective:** O predicado subobjective recebe como parâmetros **uma pessoa e um lugar**, e indica se esse lugar deverá ser visitado antes dos outros.
- **hasToMeetSomeone:** O predicado hasToMeetSomeone recebe como parâmetros **uma pessoa e um lugar**, e indica que essa pessoa deve encontrar alguém nesse lugar
- **hasToMeet:** O predicado hasToMeet recebe como parâmetros **duas pessoas e um lugar**, e indica que essas duas pessoas devem se encontrar nesse lugar.
- **isPonto:** O predicado isPonto recebe como parâmetro **um lugar**, e indica se esse lugar é um ponto turístico.

## 2.3 Ações

- **pegar-bicicleta**
  - **Descrição:** A ação pegar-bicicleta trata a ação de uma pessoa de pegar uma determinada bicicleta em uma determinada estação.
  - **Parâmetros:**
    - quem - a pessoa que quer pegar uma bicicleta
    - bicicleta - qual bicicleta a pessoa quer pegar
    - onde - estação em que a pessoa quer pegar aquela bicicleta
  - **Pré-condições:** A pessoa não deve ter uma bicicleta, a bicicleta deve estar nessa estação, a pessoa deve estar nessa estação e essa pessoa deve estar habilitada para pegar uma bicicleta.
  - **Efeito:** É identificado que aquela pessoa possui uma bicicleta, assim como qual bicicleta ela pegou. Além disso, identifica-se que a bicicleta não está mais naquela estação.
- **entregar-bicicleta**
  - **Descrição:** A ação entregar-bicicleta trata a ação de uma pessoa de entregar uma determinada bicicleta em uma determinada estação.
  - **Parâmetros:**
    - quem - a pessoa que quer entregar uma bicicleta
    - bicicleta - qual bicicleta a pessoa quer entregar
    - onde - estação em que a pessoa quer entregar aquela bicicleta
  - **Pré-condições:** A pessoa deve estar com uma bicicleta, e essa bicicleta deve ser a bicicleta que se deseja entregar e a pessoa e a bicicleta devem estar na estação passada como parâmetro.
  - **Efeito:** Será identificado que a pessoa não possui uma bicicleta, que aquela

peessoa não pode mais pegar uma bicicleta (ela deve esperar 5 minutos) e identifica que a bicicleta que foi entregue não tem mais seu tempo de pedalada expirado.

- **esperar5min**

- **Descrição:** A ação esperar5min trata a ação de uma pessoa de esperar 5 minutos na estação para poder ser habilitada de pegar uma bicicleta.
- **Parâmetros:**
  - quem - a pessoa que vai esperar 5 minutos.
- **Pré-condições:** A pessoa não pode ter uma bicicleta para realizar essa ação.
- **Efeito:** A pessoa será habilitada de pegar uma bicicleta.

- **caminhar**

- **Descrição:** A ação caminhar trata a ação de uma pessoa de ir de uma estação até um ponto turístico ou vice-versa.
- **Parâmetros:**
  - quem - a pessoa que irá caminhar.
  - lugar1 - o local que a pessoa se encontra no momento.
  - lugar2 - o local para onde a pessoa irá caminhar.
- **Pré-condições:** A pessoa deve estar no primeiro local especificado, não deve estar com uma bicicleta e os locais precisam ser adjacentes.
- **Efeito:** Será recordado que a pessoa está agora no segundo local, e não mais no primeiro.

- **visitar-ponto**

- **Descrição:** A ação visitar-ponto trata a ação de uma pessoa entrar (fazer visita) em um determinado ponto turístico.
- **Parâmetros:**
  - quem - a pessoa que fará a visita.
  - atracao - ponto turístico que a pessoa fará a visita.
- **Pré-condições:** A pessoa deve estar no local do ponto turístico especificado, esse ponto deve ser um ponto turístico e a pessoa não pode ter pendências para visitar outro local antes, ou se possui dependência, essa já foi completada. Além disso, ela não poderá visitar se no local ela necessita esperar por alguém.
- **Efeito:** Será recordado que essa pessoa visitou tal lugar, ela será habilitada para pegar uma bicicleta quando eventualmente voltar para uma estação e caso essa visita seja uma pendência para outras visitas, será marcada como

concluída.

- **pedalar**

- **Descrição:** A ação pedalar trata a ação de uma pessoa ir de uma estação a outra.
- **Parâmetros:**
  - quem - a pessoa que irá pedalar
  - bicicleta - a bicicleta que a pessoa está usando
  - estacao1 - a estação que a pessoa se encontra.
  - estacao2 - a estação para onde a pessoa irá pedalar.
- **Pré-condições:** A pessoa deve estar na primeira estação especificada, deve estar com uma bicicleta, essa bicicleta não deve ter seu tempo de pedalada expirado e essa bicicleta deve ser a bicicleta especificada. Além disso, as estações precisam ser adjacentes.
- **Efeito:** Será recordado que a pessoa está agora na segunda estação, e não mais na primeira. Além disso, o tempo de pedalada da bicicleta irá expirar e a localização da bicicleta será atualizada para a segunda estação.

- **se-encontrar**

- **Descrição:** A ação se-encontrar trata a ação de uma pessoa se encontrar com outra em algum local.
- **Parâmetros:**
  - quem1 - primeira pessoa que irá se encontrar com outra
  - quem2 - segunda pessoa que irá se encontrar com a outra
  - lugar - o local onde as pessoas se encontrarão.
- **Pré-condições:** As duas pessoas devem estar no local especificado e essas duas pessoas devem ter pendência de se encontrar no local especificado.
- **Efeito:** Será recordado que as duas pessoas não precisam mais se encontrar com ninguém naquele local.

## 2.4 Restrições

Existem duas restrições principais para o problema:

1. João deve visitar o Banco do Brasil antes de visitar o Paço Alfândega
2. Maria e José devem se encontrar no Mercado de São José antes de visitá-lo

A **primeira restrição** é garantida declarando:

*(hasSubobjective joao)*

*(subobjective joao banco-brasil)*

pois dessa forma sempre que o planejador tentar usar **visitar-ponto** para João em uma atração, será checado se a atração é o sub-objetivo. Caso não seja, João será impedido de visitar esse ponto até visitar o sub-objetivo (Banco do Brasil).

Já a **segunda restrição** é garantida declarando:

*(hasToMeetSomeone jose mercado)*  
*(hasToMeetSomeone maria mercado)*  
*(hasToMeet jose maria mercado)*

Dessa forma toda vez que o planejador tentar visitar uma atração ele irá checar se a pessoa visitando deve se encontrar com alguém nessa atração e não permitirá que ela visite o local até que os dois estejam nesse lugar. Estando os dois no mesmo lugar, o planejador poderá chamar a action **se-encontrar** e assim fará com que as condições

**hasToMeetSomeone** passem a ser falsas e que o dois possam enfim visitar a atração. A condição **hasToMeet jose maria mercado** garante que o encontro será entre José e Maria.

## 2.5 Objetivo

O objetivo é definido pelo predicado **hasBeen** indicando o local que cada um das pessoas deve visitar antes que o plano termine, além disso é usado o predicado **withBike** para que nenhum dos turistas tenha uma bicicleta no fim, obrigando-os a devolvê-las.



### 3. Resultados

Executando o comando:

```
ff -o turismo-domain.pddl -f turismo-problem.pddl
```

Obtivemos o seguinte resultado na execução do planejamento:

```
ff: found legal plan as follows
step    0: PEGAR-BICICLETA JOSE BIKE-8 BIKE-APOLO
        1: PEGAR-BICICLETA JOAO BIKE-6 BIKE-DIARIO
        2: PEDALAR JOSE BIKE-8 BIKE-APOLO BIKE-REPUBLICA
        3: ENTREGAR-BICICLETA JOSE BIKE-8 BIKE-REPUBLICA
        4: PEDALAR JOAO BIKE-6 BIKE-DIARIO BIKE-SANTA-RITA
        5: CAMINHAR JOSE BIKE-REPUBLICA PRACA-REPUBLICA
        6: VISITAR-PONTO JOSE PRACA-REPUBLICA
        7: ENTREGAR-BICICLETA JOAO BIKE-6 BIKE-SANTA-RITA
        8: ESPERAR5MIN JOAO
        9: CAMINHAR JOSE PRACA-REPUBLICA BIKE-REPUBLICA
       10: PEGAR-BICICLETA JOAO BIKE-6 BIKE-SANTA-RITA
       11: PEDALAR JOAO BIKE-6 BIKE-SANTA-RITA BIKE-ALFANDEGA
       12: PEGAR-BICICLETA JOSE BIKE-8 BIKE-REPUBLICA
       13: PEDALAR JOSE BIKE-8 BIKE-REPUBLICA BIKE-DIARIO
       14: ENTREGAR-BICICLETA JOAO BIKE-6 BIKE-ALFANDEGA
       15: ESPERAR5MIN JOAO
       16: PEGAR-BICICLETA JOAO BIKE-6 BIKE-ALFANDEGA
       17: PEDALAR JOAO BIKE-6 BIKE-ALFANDEGA BIKE-APOLO
       18: ENTREGAR-BICICLETA JOAO BIKE-6 BIKE-APOLO
       19: CAMINHAR JOAO BIKE-APOLO BANCO-BRASIL
       20: VISITAR-PONTO JOAO BANCO-BRASIL
       21: CAMINHAR JOAO BANCO-BRASIL BIKE-APOLO
       22: PEGAR-BICICLETA JOAO BIKE-6 BIKE-APOLO
       23: PEDALAR JOAO BIKE-6 BIKE-APOLO BIKE-ALFANDEGA
       24: ENTREGAR-BICICLETA JOAO BIKE-6 BIKE-ALFANDEGA
       25: CAMINHAR JOAO BIKE-ALFANDEGA PACO-ALFANDEGA
       26: VISITAR-PONTO JOAO PACO-ALFANDEGA
       27: PEGAR-BICICLETA MARIA BIKE-6 BIKE-ALFANDEGA
       28: PEDALAR MARIA BIKE-6 BIKE-ALFANDEGA BIKE-SANTA-RITA
       29: ENTREGAR-BICICLETA JOSE BIKE-8 BIKE-DIARIO
       30: ESPERAR5MIN JOSE
       31: PEGAR-BICICLETA JOSE BIKE-8 BIKE-DIARIO
       32: PEDALAR JOSE BIKE-8 BIKE-DIARIO BIKE-MERCADO
       33: ENTREGAR-BICICLETA JOSE BIKE-8 BIKE-MERCADO
       34: CAMINHAR JOSE BIKE-MERCADO MERCADO
       35: ENTREGAR-BICICLETA MARIA BIKE-6 BIKE-SANTA-RITA
       36: ESPERAR5MIN MARIA
       37: PEGAR-BICICLETA MARIA BIKE-6 BIKE-SANTA-RITA
       38: PEDALAR MARIA BIKE-6 BIKE-SANTA-RITA BIKE-DIARIO
       39: ENTREGAR-BICICLETA MARIA BIKE-6 BIKE-DIARIO
       40: CAMINHAR MARIA BIKE-DIARIO IGREJA
       41: VISITAR-PONTO MARIA IGREJA
       42: CAMINHAR MARIA IGREJA BIKE-DIARIO
       43: PEGAR-BICICLETA MARIA BIKE-6 BIKE-DIARIO
       44: PEDALAR MARIA BIKE-6 BIKE-DIARIO BIKE-MERCADO
       45: ENTREGAR-BICICLETA MARIA BIKE-6 BIKE-MERCADO
       46: CAMINHAR MARIA BIKE-MERCADO MERCADO
       47: SE-ENCONTRAR JOSE MARIA MERCADO
       48: VISITAR-PONTO JOSE MERCADO
       49: VISITAR-PONTO MARIA MERCADO

time spent: 0.01 seconds instantiating 898 easy, 15 hard action templates
            0.00 seconds reachability analysis, yielding 198 facts and 673 actions
            0.00 seconds creating final representation with 151 relevant facts
            0.00 seconds building connectivity graph
            0.01 seconds searching, evaluating 403 states, to a max depth of 3
            0.02 seconds total time
```

Dessa forma é possível constatar que os objetivos foram cumpridos e os pontos foram visitados na seguinte ordem:

- **João:** Banco do Brasil (passo 20) e Paço Alfândega (passo 26)
- **José:** Praça da República (passo 6) e Mercado de São José (passo 49)
- **Maria:** Igreja (passo 41) e Mercado (passo 49)

É possível verificar também que as restrições foram satisfeitas, pois João visitou o Banco do Brasil antes de visitar o seu outro objetivo e José chegou ao mercado e não o visitou até Maria também chegar nele, para assim os dois se encontrarem e visitarem o local juntos.