

Um habitante extra-terrestre pode ser representado através da seguinte classe:

```
class Alien{
    string nome;
    int energia;
public:
    Alien(string s):nome(s), energia(10){}

    string getNome() const {return nome;}
    int getEn() const {return energia;}
};
```

Durante a resposta a esta questão, não pode alterar o que está feito na classe *Alien*. Caso seja relevante, poderá adicionar atributos e/ou métodos.

Um Sistema Planetário é constituído por um conjunto de planetas. Por sua vez, os planetas abrigam um conjunto de habitantes extra-terrestres. O sistema planetário toma posse dos seus planetas e os planetas são os únicos responsáveis pelos habitantes que aí se encontram. Implemente as classes *SistemaPlanetario* e *Planeta*, de forma a que o código seguinte seja executado corretamente:

```
int main(){

    // Cria um Sistema planetário com 2 planetas
    SistemaPlanetario OuterRim({new Planeta("Tatooine"), new
Planeta("Endor")});

    // Adiciona extra-terrestres aos planetas do Sistema Planetário.
    //Primeiro verifica se existem
    if(OuterRim.existePlaneta("Tatooine"))
        OuterRim.addAlien(new Alien("Greedo"), "Tatooine");

    if(OuterRim.existePlaneta("Endor"))
        OuterRim.addAlien(new Alien("Jabba"), "Endor");

    // Move um extra-terrestre de um planeta para outro
    OuterRim.mover("Chewbacca", "Tatooine", "Endor");

    // Mostra informação detalhada sobre os planetas e habitantes do
    //sistema planetário
    cout << OuterRim << endl;

    return 0;
}
```

- Pode assumir que os nomes dos planetas e dos aliens são únicos.
- Os métodos para adicionar alien e para mover alien só efetuam as operações, caso os aliens e planetas envolvidos existam. Estes métodos são booleanos e devolvem informação sobre o resultado da operação.
- O operador de atribuição não está disponível para objetos do tipo *SistemaPlanetario* (e apenas para este tipo de objetos). Todas as restantes operações devem poder ser efetuadas da forma prevista.