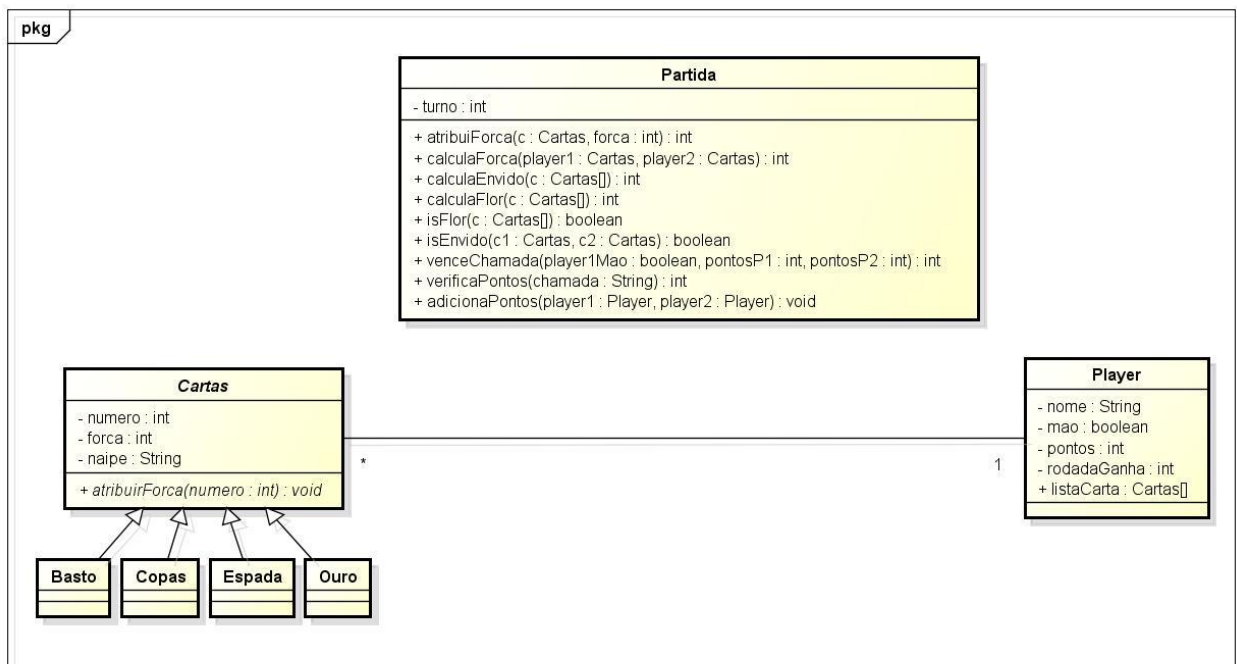


# trucOO

## Tecnologia de Orientação a Objetos

Diego Soria Rios e Paulo Cesar Martins Citron



## **Descrição**

A organização baseia-se em classes de Naipes originadas de uma classe abstrata mãe Cartas, com uma classe Player para os jogadores (onde são armazenadas as cartas, os pontos e as informações do player) e uma classe para a partida (onde são armazenados os métodos usados para cálculos do jogo, como calcular um envio, por exemplo).

## **Encapsulamento**

Utilizar getters e setters deixa o código seguro além de manipular os dados com mais facilidade.

## **Abstração**

Utilizado no código para gerar uma carta, e posteriormente evadida em quatro naipes, ajudou na manipulação da força de cada carta (com atribuiValor()), deixando coisas usuais em todos os naipes na classe Cartas original, tornando mais fácil dividir a força, que seria diferente em determinados naipes, sem reescrever código.

Optou-se por utilizar o método atribuiForca() como métodos abstratos dentro de cada classe, o que facilitou a construção do código.

## **Herança**

As quatro classes filhas da classe abstrata Cartas, ajudaram na seleção de força para cada naipe e determinado número, tornando o código mais organizado.

## **Polimorfismo**

Utilizado para criar os objetos de diferentes classes quando as cartas são dadas.

## **Orientação a Objetos**

A reutilização de código é um dos principais fatores positivos do uso de Pilares da Orientação a Objetos. Outro grande ponto positivo no uso da Orientação a Objetos é que o código possui uma leitura mais nítida, e um desenvolvimento mais rápido por seu fácil entendimento.

## **Dificuldades**

Algumas dificuldades como o tamanho e complexidade do código, por se tratar de um jogo com muitas regras e sub regras, foram facilmente contornadas usando a Orientação a Objetos.

## **Conclusão**

Temos um jogo de truco para dois jogadores, sendo possível realizar todas as jogadas de um jogo real. Com o código organizado e de fácil compreensão do funcionamento do programa, podemos concluir que a Orientação a Objetos é uma grande ferramenta para desenvolvimento de programas que ofertam muitas regras e demandam performance.