

# Computação II – Orientação a Objetos

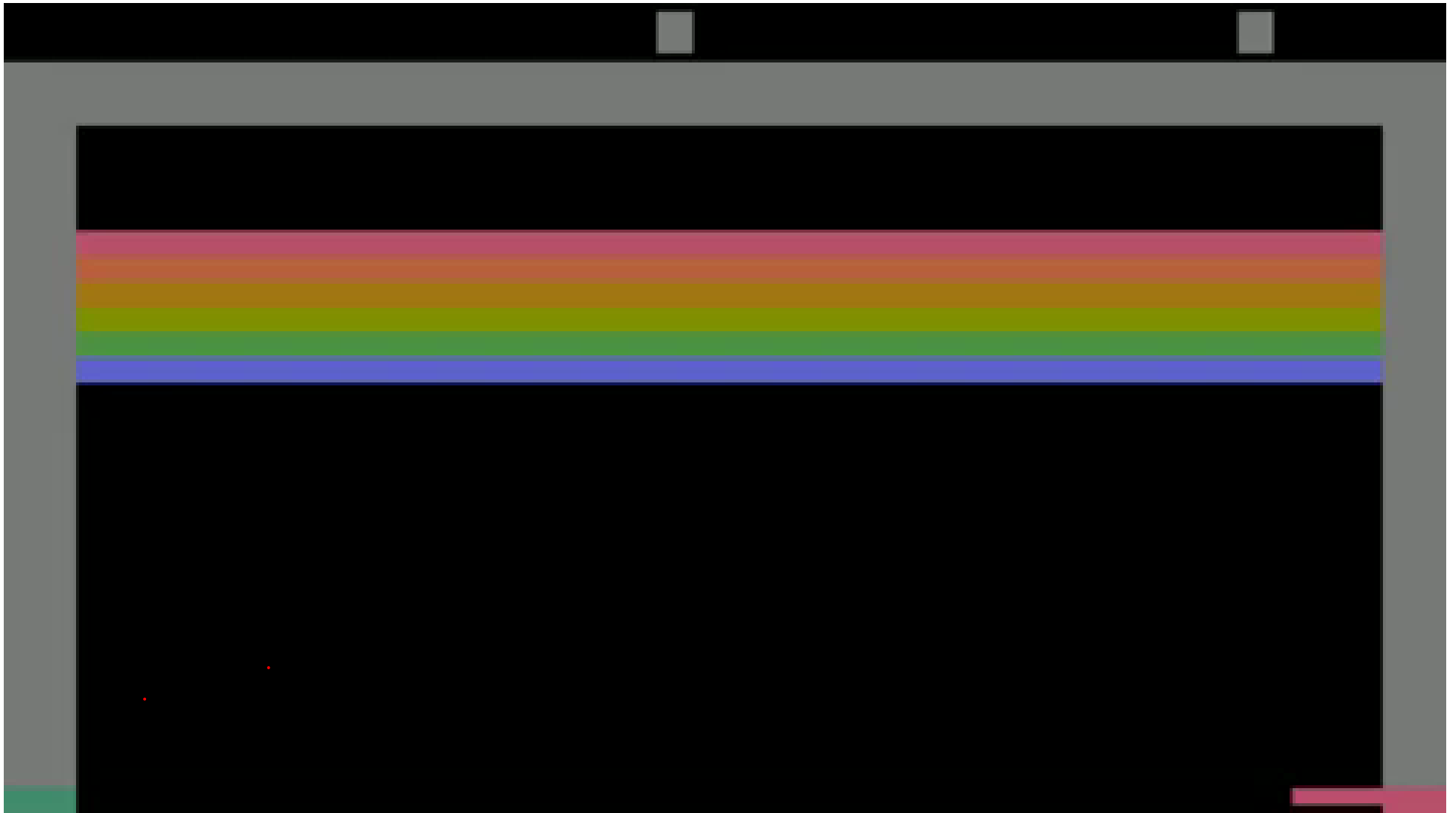
---

Fabio Mascarenhas - 2014.1

<http://www.dcc.ufrj.br/~fabiom/java>

# Breakout

---



# Componentes do Breakout

---

- Bola
- "Raquete"
- Tijolos
- Paredes *→ campos na classe Jogo*
- Score *e vidas → campos na classe Jogo*
- Nem todos vão precisar de classes próprias para representá-los!

# Laço for de coleções

---

- Java tem uma versão do laço for que é especializada para percorrer uma coleção (um vetor, ou instâncias de uma das classes de coleção como HashSet e ArrayList)
- O bloco do laço é executado para cada elemento da coleção, com a variável de controle apontando para o elemento

```
for(int i: vetor) {  
    System.out.println(i);  
}
```

# Coordenação

---

- Tanto no breakout como em outros jogos, precisamos verificar possíveis colisões entre os objetos do jogo, e tomar ações a depender de qual objeto colidiu com qual
  - Ex: *se a bola colide com um tijolo, o tijolo some e a bola é refletida*
- Verificar uma interação envolvendo campos de dois (ou mais) objetos diferentes, e disparar ações em todos eles, é um problema da modelagem OO
- De quem é a responsabilidade de *coordenar* essa interação?

# Coordenação, cont.

---

- Podemos eleger um dos objetos que estão participando da interação para ser o coordenador, mas isso aumenta o *acoplamento* entre os objetos que estão interagindo
- Ou podemos usar um [mediador](#), um objeto que vai verificar se houve alguma interação, e mandar os objetos envolvidos tomarem uma ação
- O mediador precisa ter acesso ao estado dos objetos que estão interagindo, mas o acoplamento entre esses objetos diminui
- Vamos usar a instância de Jogo como mediador em nosso exemplo