

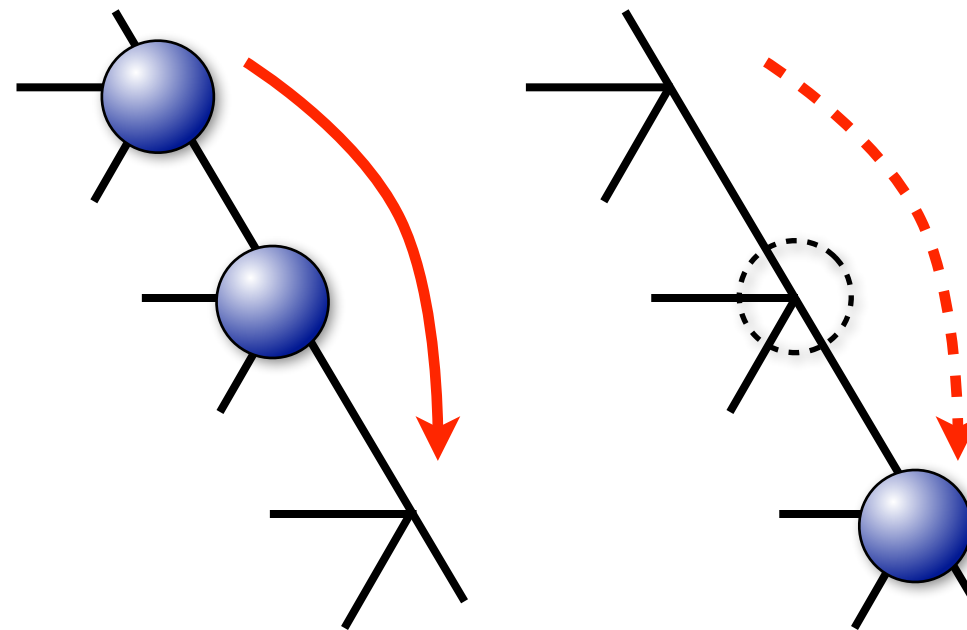
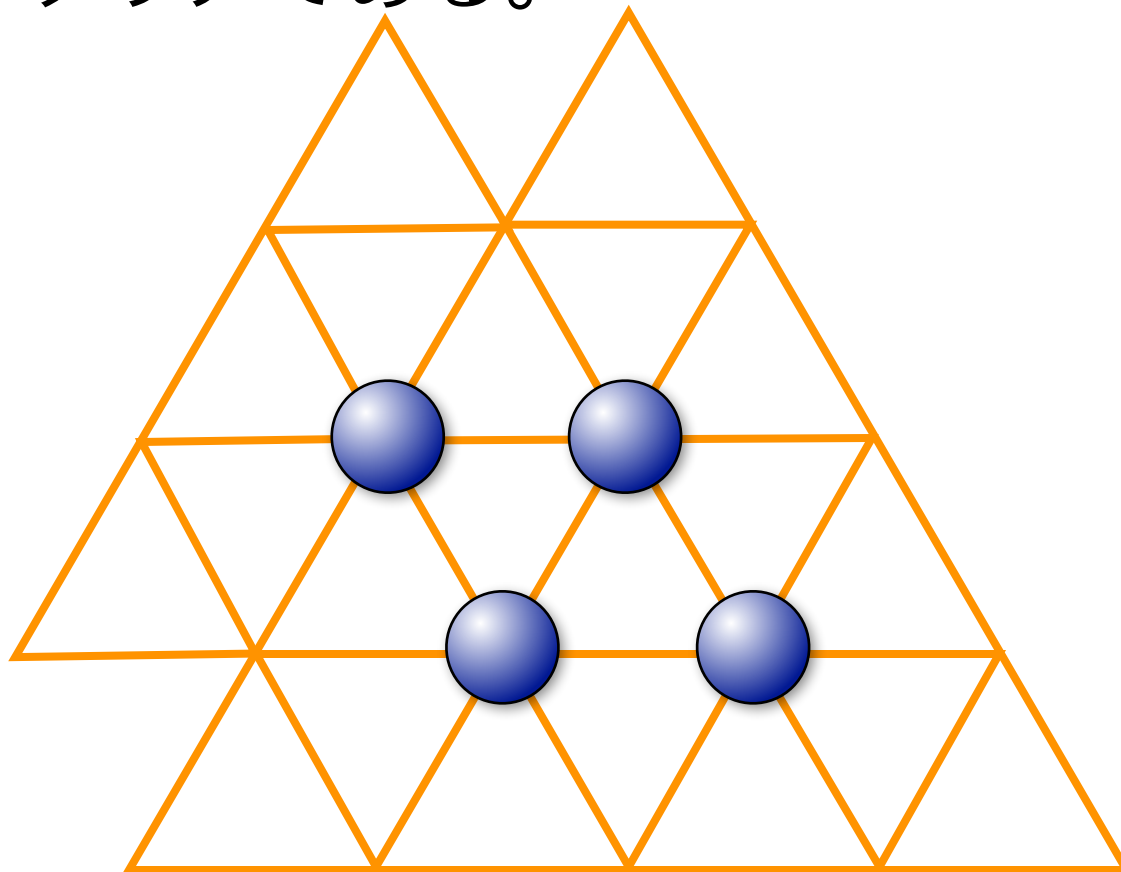
# ペグ・ソリティア

2014年 プロジェクト演習 第2期（荻原）

Version 2.0 (2014.05.30)

# ルールの概要

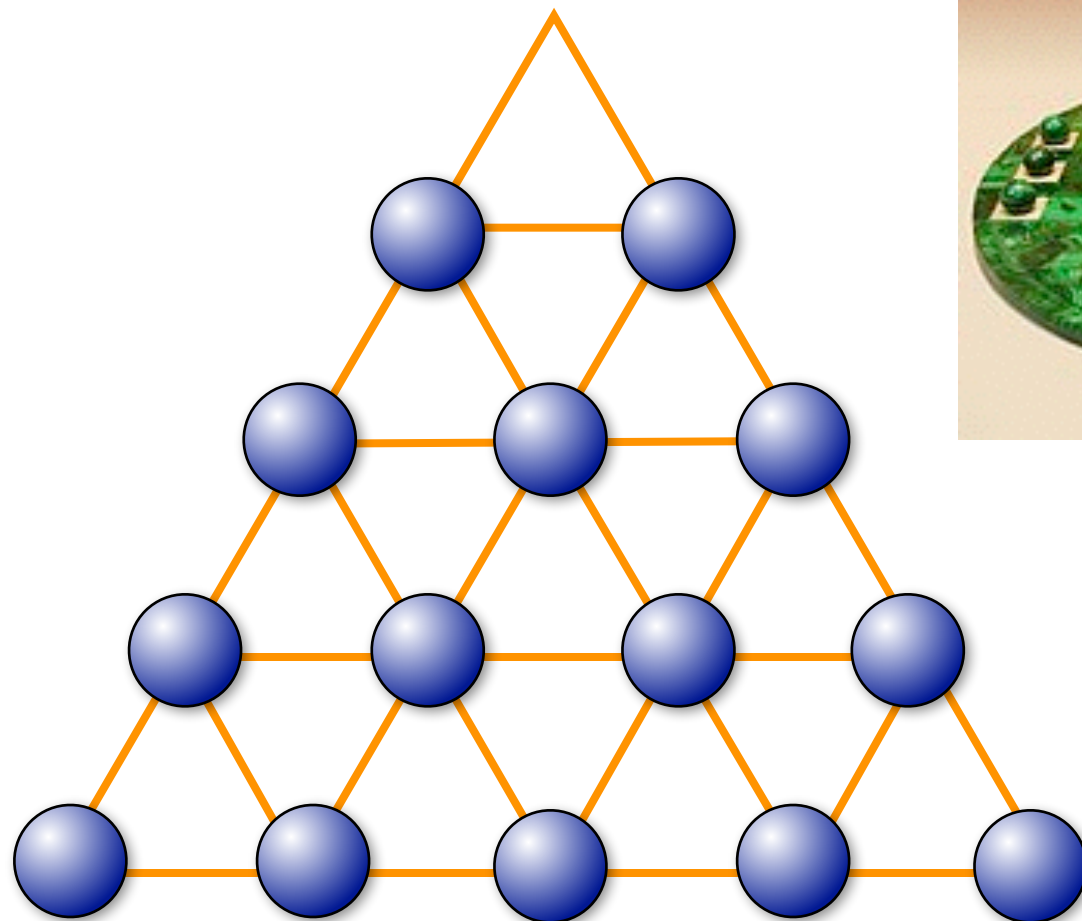
正三角形（正方形）が集まって構成されたボードがある。  
頂点のいくつかにペグ（杭や釘のこと）を置く。  
ペグは、隣接するペグを飛び越して空いた頂点に移動することができ、この時、飛び越されたペグは消すことができる。  
それ以外にペグを動かすことはできない。  
この繰り返しで、最後にペグをひとつだけにできればゲームクリアである。



# ペグ・ソリティア

このゲームは、ペグ・ソリティア Peg Solitaire  
と言う。

例えば、下の例がよく  
知られている。



よく知られた別のタイプ



[http://en.wikipedia.org/wiki/Peg\\_solitaire](http://en.wikipedia.org/wiki/Peg_solitaire)

# ペグ・ソリティアの実装例

ここに示したのは、Flashによる実装。



[http://www.gamedesign.jp/flash/peg/peg\\_jp.html](http://www.gamedesign.jp/flash/peg/peg_jp.html)

# ゲームの流れ

- ▶ 最初に、ボードを表示し、ペグをいくつか置く。
- ▶ ユーザは動かしたいペグと動かす先の位置を指示する。飛び越されたペグは消える。
- ▶ 移動を繰り返して、ペグが1つだけ残ったら成功。
- ▶ そうでない場合は、やり直しか放棄。

# 要求

- ◆ ペグ・ソリティアとして遊べるアプリケーションを、Java言語を使って実装する。
- ◆ ボードは格子状（または正方形が集まった形）を基本とする（斜めには移動できないとする）。
- ◆ ちゃんと解ける問題（初期配置）を複数個用意すること。問題はソースコードとしてコーディングするのではなく、データファイルにしておき、実行中に読み込んで動作させる。
- ★ 可能であれば、問題（ペグの初期配置）の作成を支援する機能も実現する。
  - ➡ 機能を付加するか、別のアプリとして作成

# 注意

- 詳細な画面表示や操作方法は自由。作りやすいように（ゲームが面白いように）すればよい。
- データファイルの形式も自由。
- 最初からやり直す、または、ひとつ後戻りする方法は必要だろう。
- 通れない場所を作る、ヒントを示す、などのバリエーションを加えてもよい。
- 完成までの秒数を表示してもよい。

# 実装に関するヒント

- モデル、ビュー、コントローラ（MVC）の考え方で実装しよう
  - ◆ ボードを表示し、入力を検知する「ビュー」を用意する
  - ◆ 入力と表示、メモリ内の情報の変更の仲介を行う「コントローラ」が必要
  - ◆ 盤面情報の保持・更新を行う「モデル」が必要

**注意！** 進捗状況を見て、条件の追加、変更を指示することがあります。