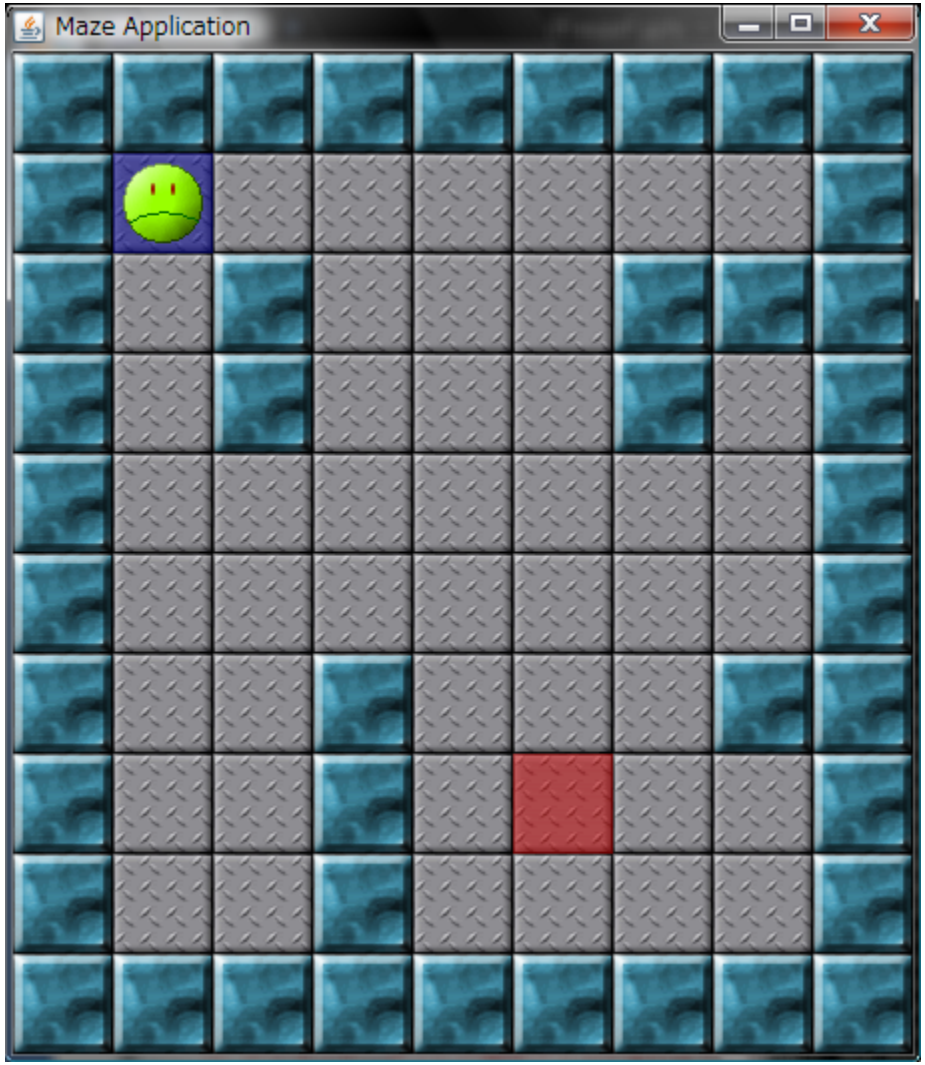
CS 知的システム演習

迷路を解くプログラム

松吉 俊

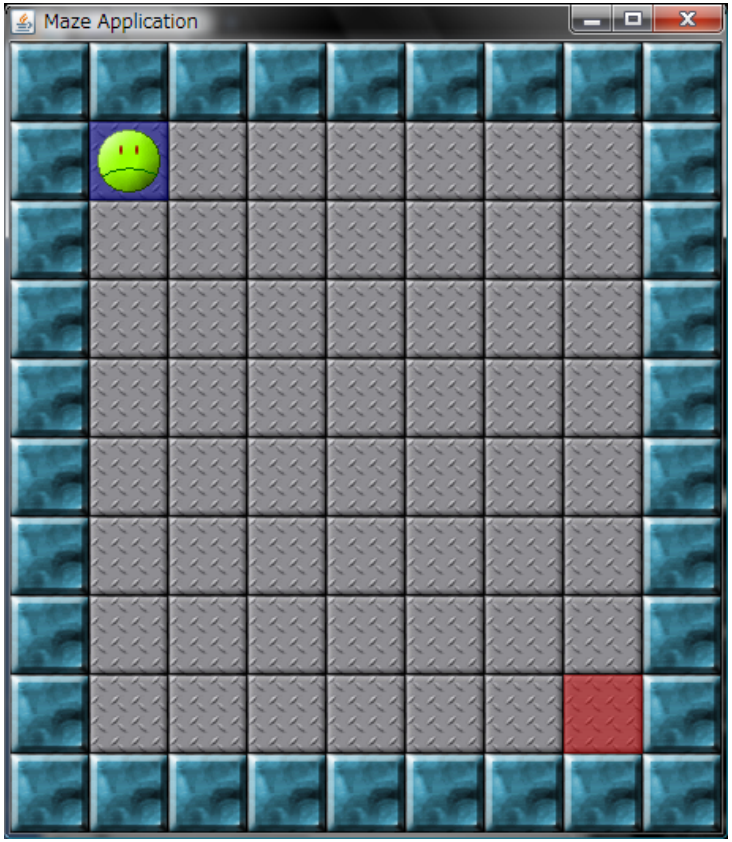
迷路を解くプログラムの製作

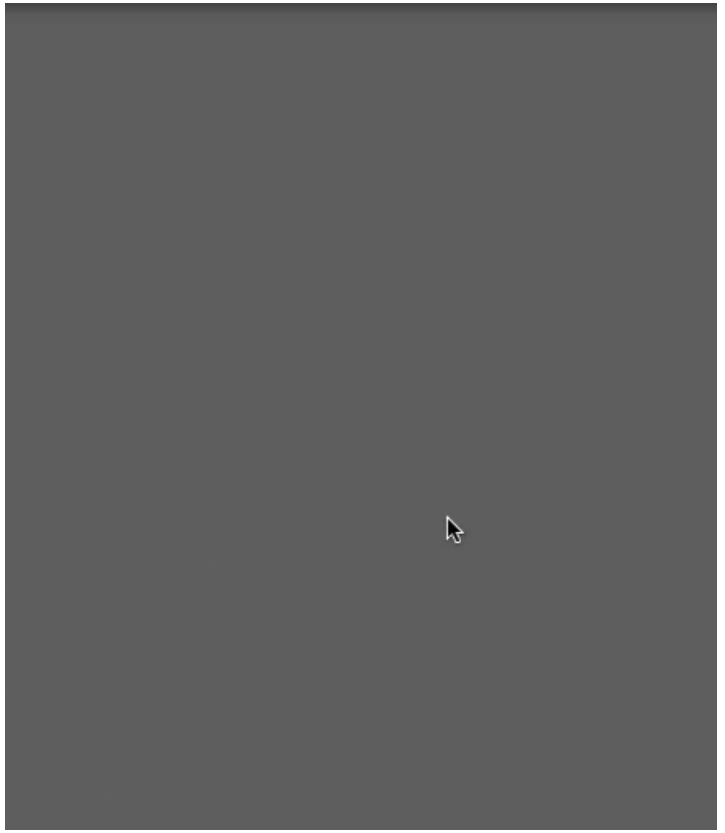


- 青マス：　スタート位置
- 赤マス：　ゴール位置
- 通路: 通れます
- 壁: 通れません
- ロボットの行動
  - 上下左右に1マス移動
  - 斜め移動は不可
  - 2マス以上先への瞬間移動も不可

デモ

- 雛形プログラムのデモ





使用するプログラミング言語

- Java言語
  - 『ソフトウェア設計開発演習I』で学習済み
  - 基本的な能力があれば十分
    - 継承などの難しい概念は利用しません
  - 雛形プログラムを提供
    - 本質的な部分のみ作成すればよい
  - 分からないことがあったら、
    まずは上記授業の資料を復習
    - TAにも気軽に質問してください

雛形プログラム

ソースファイル

Maze.java	起動用のクラス
MazeData.java	迷路データを表すクラス
Robot.java	ロボットを表すクラス
MazeModel.java	ロボットを制御するクラス
MazeView.java	描画を担当するクラス

基本的にはこのクラスのみ編集すれば十分

※ その他のクラスも自由に編集して良い

画像ファイル

- block.png
- goal.png
- robot1.png
- robot2.png
- space.png
- start.png

マップファイル

- map1.txt
- map2.txt
- map3.txt

雛形プログラムのダウンロード

1. Moodleから、maze-template.tar.gz をダウンロードする
2. 適当なディレクトリに保存する
3. 圧縮ファイルを解凍する

```
% tar xvzf maze-template.tar.gz
```

UNIXコマンドに不安がある場合は、
KKI Living Guideなどを参照してください

雛形プログラムのコンパイルと実行

- コンパイル

  % javac *.java ← すべての.javaファイルから、.classファイルが生成されます

- 実行

  % java Maze map1.txt

  起動用のメインclass

※ map1.txtを解く場合。
map2.txtやmap3.txtを解く場合も同様に実行します

```
/**
 * 実行用関数
 */
public void run()
{
  try {
    int gx = mazeData.getGX();
    int gy = mazeData.getGY();

    while (true) {

      int x = robot.getX();
      int y = robot.getY();

      if (mazeData.get(x、y+1) != MazeData.BLOCK)
        y++;

      robot.setX(x);
      robot.setY(y);

      mazeView.repaint();

      Thread.sleep(500);

      if (x == gx && y == gy)
        break;
    }
  } catch (Exception e) {
    e.printStackTrace();
    System.exit(-1);
  }
}
```

MazeModel.java 内にある関数

// ゴール座標の取得

// ロボットの現在位置を取得

// 下に行けるならば下に行く

// ロボットの位置座標を更新

// 現在の状態を描画する

// 速すぎるので 500msec 寝る

// もしゴールに到達すれば終了

ここをがんばって作る

クラス MazeData の概要

- メソッド
  - get(int x、int y)　指定座標 (x、y) にあるものを返す
    - MazeData.SPACE　通路
    - MazeData.BLOCK　壁
    - MazeData.START　スタート
    - MazeData.GOAL　ゴール
  - getSX()　スタート位置の X 座標を返す
  - getSY()　スタート位置の Y 座標を返す
  - getGX()　ゴール位置の X 座標を返す
  - getGY()　ゴール位置の Y 座標を返す
  - getWidth()　マップの横サイズを返す
  - getHeight()　マップの縦サイズを返す

クラス Robot の概要

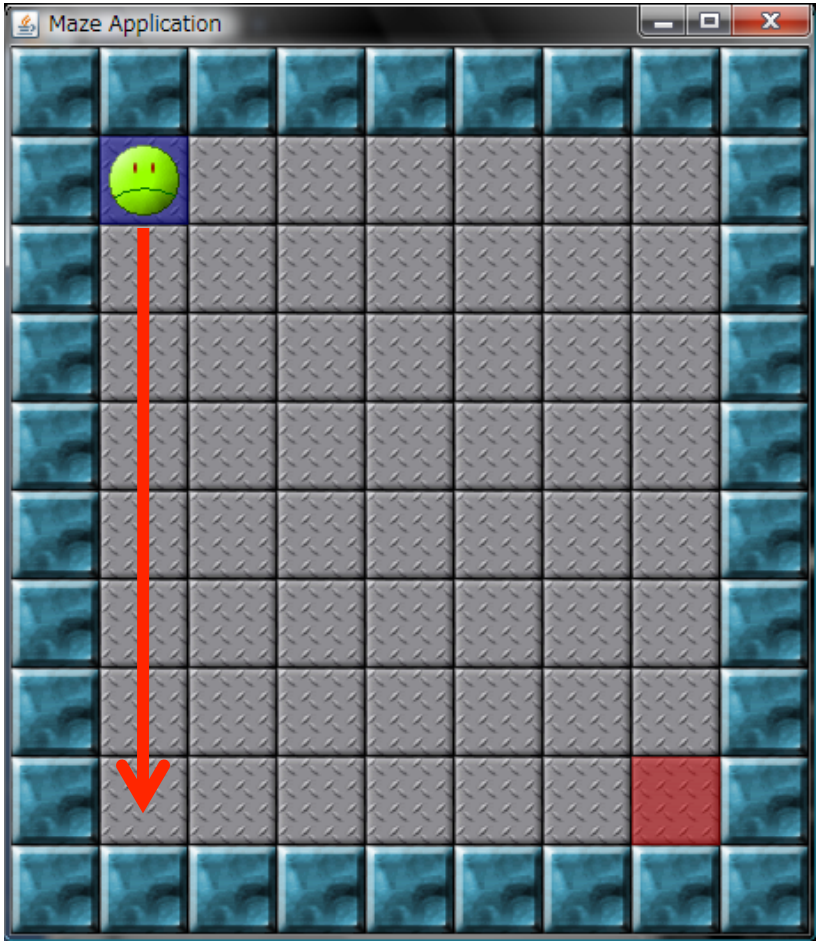
- メソッド
  - getX()　　ロボットの X 座標を返す
  - getY()　　ロボットの Y 座標を返す
  - setX(int x)　　ロボットの X 座標を x に設定する
  - setY(int y)　　ロボットの Y 座標を y に設定する

ロボットは1マスずつ移動させること！

演習1

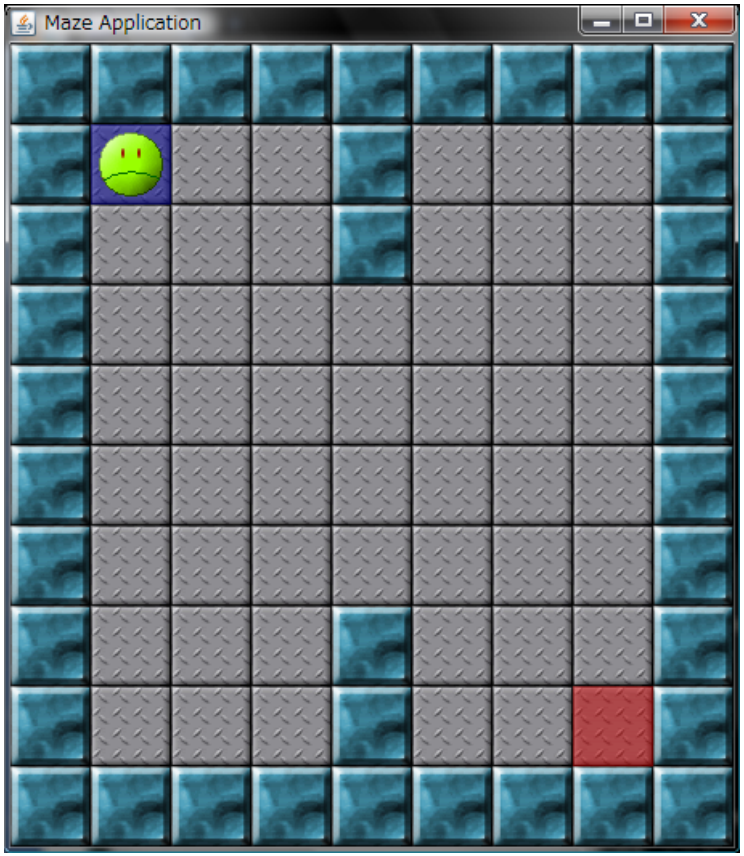
- map1.txt を解くアルゴリズムを考案し、実装せよ

雛形では、ぶつかるまで
下に移動する



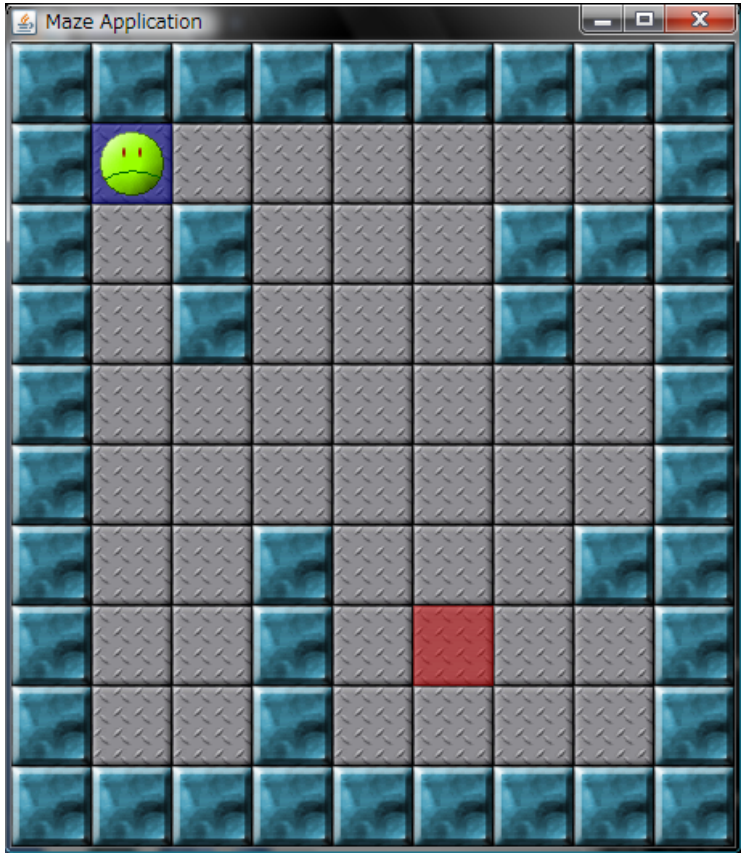
演習2

- map1.txt だけでなく、map2.txtも解くアルゴリズムを考案し、実装せよ



演習3

- アルゴリズムを汎用化し、map3.txt も 解くことができるように改良せよ



第1回の出席確認

- 実装できたところまでで良いので、演習3 (or 演習2 or 演習1)に対するプログラムをMoodle上で提出する
  - tar.gz形式のファイルを提出する
  - 最終評定には直接関係しません

- Special thanks:
  - 山本 泰生先生
  - 鍋島 英知先生