

泥棒シミュレーション

～泥棒大作戦!!～

プログラミング演習 B グループ：平岩 健一郎

Kenichiro HIRAIWA

1 はじめに

オートマトンとは、計算機における「状態」と「状態遷移」から成る問題解決手順を、数学的な視点からモデリングして定式化したものであり、コンパイラの字句解析、Web などのテキストからのキーワード検索などに応用されている。

今回、オートマトンの概念を用いて「泥棒が家に入り込んで金庫を開けてお宝を手に入れるシミュレーション」を作成する。

2 泥棒シミュレーションのアルゴリズム

状態は泥棒の所持アイテムの組み合わせとし、状態遷移をアイテムの入手・使用とする。泥棒シミュレーションのオートマトンを Fig. 1 に示す。各ノードの内部は「状態: 所有しているアイテム, 所有しているアイテム, ...」を示している。本システムでは、泥棒は、5つのアイテムのパラメータ「手袋, 針金, 本, 鍵, お宝」を持ち、アイテムを所有している場合は1, いない場合は0で表す。

本システムのストーリーは大きく分けて、以下の3段階に分けられる。

- ストーリー1. 扉を開けて部屋に入る
- ストーリー2. 金庫の鍵を手に入れるため、4つの部屋を探し回る
- ストーリー3. 手に入れた鍵を使用し金庫を開ける

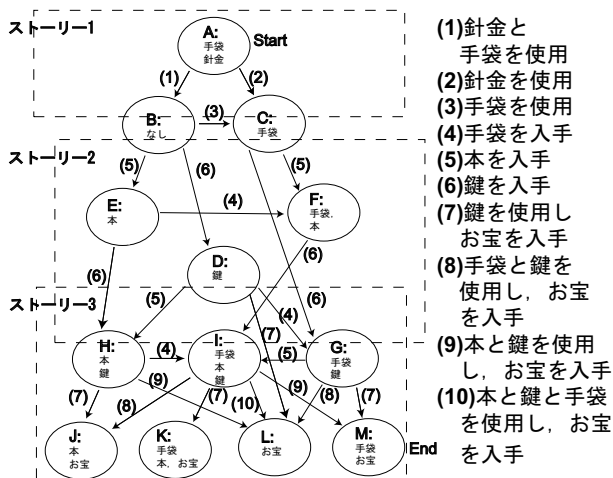


Fig. 1 泥棒シミュレーションの状態遷移図

2.1 扉を開けて部屋に入る (ストーリー 1)

シミュレーションのスタートの状態はAであり、このとき「泥棒」は「手袋」と「針金」を持っている。状態Aは入口にあたり、ここから扉を開けて家の中に入る。状態Aで泥棒が使うアイテムの選択肢としては「手袋と針金」「針金」「手袋」の3通りある。このストーリー1の遷移規則を Table 1 に示す。現状態, 「後」は次状態, 「遷移」は遷移条件, 「番号」は Fig. 1 における遷移番号を示している。

Table 1 ストーリー 1

| 前 | 後 | 遷移 | 番号 |
|---|---|----------|-----|
| A | B | 針金と手袋を使用 | (1) |
| A | C | 針金を使用 | (2) |

2.2 金庫の鍵を探し回る (ストーリー 2)

(1), (2) の状態遷移により, 「泥棒」は状態 B, または C の状態になる。この状態から、ミニゲームをしアイテムを入手することにより、次の状態に遷移していく。ミニゲームにより入手できるアイテムは「手袋」, 「鍵」, 「本」である。Fig. 1 における、ストーリー 2 の遷移規則を Table 2 に示す。

Table 2 ストーリー 2

| 前 | 後 | 遷移 | 番号 |
|---|---|-------|-----|
| B | C | 手袋を入手 | (3) |
| B | D | 鍵を入手 | (6) |
| B | E | 本を入手 | (5) |
| C | F | 本を入手 | (5) |
| C | G | 鍵を入手 | (6) |
| D | G | 手袋を入手 | (4) |
| D | H | 本を入手 | (5) |
| E | F | 手袋を入手 | (4) |
| E | H | 鍵を入手 | (6) |
| F | I | 鍵を入手 | (6) |
| G | I | 本を入手 | (5) |
| H | I | 手袋を入手 | (4) |

2.3 金庫を開ける (ストーリー 3)

泥棒が金庫の鍵を手に入れたら、宝物庫において金庫を開けることが可能になる。お宝を手に入れた状態 J, K, L, M になるとエンドとなり終了する。Fig. 1 におけるストーリー 3 の遷移規則を Table 3 に示す。

3 実際の動作

スタート画面を Fig. 2 に示す。この画面には「入口」ボタンと「やっぱ逃げよう」ボタンがあり、「入口」ボタ

Table 3 ストーリー 3

| 前 | 後 | 遷移 | 番号 |
|---|---|------------------|------|
| D | L | 鍵を使用しお宝を入手 | (7) |
| G | L | 手袋と鍵を使用しお宝を入手 | (8) |
| G | M | 鍵を使用しお宝を入手 | (7) |
| H | J | 鍵を使用しお宝を入手 | (7) |
| H | L | 本と鍵を使用しお宝を入手 | (9) |
| I | J | 手袋と鍵を使用しお宝を入手 | (8) |
| I | K | 鍵を使用しお宝を入手 | (7) |
| I | L | 本とかぎと手袋を使用しお宝を入手 | (10) |
| I | M | 本と鍵を使用しお宝を入手 | (9) |

ンを押すとゲーム開始、「やっば逃げよう」ボタンを押すと、Game Over になる。また、制限時間を設け、残り時間が 0 になると同様に Game Over になる。

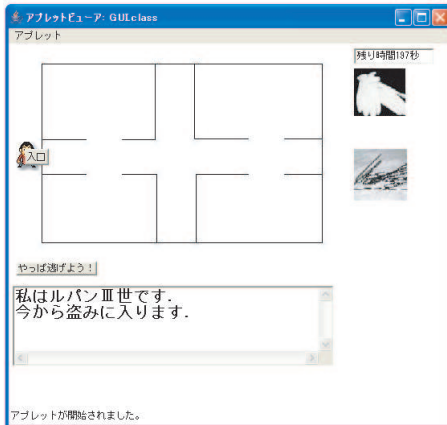


Fig. 2 スタート画面

3.1 扉を開けるときの動作

Fig. 2 の「入口」ボタンを押すと Fig. 3 のようにマップが扉の画像に切り替わり、アイテム欄に「チェックボックス」と「決定」ボタンが表示される。好きなものを選択し、「決定」ボタンを押すと Fig. 4 に示す家の中に入ることができる。ただし、2.1 節に述べた通り、針金を選ばなければ先へは進めない。



Fig. 3 アイテムの選択と決定

3.2 4つの部屋を回る動作

扉を開け家の中に入ると Fig. 4 のように「訓練所」「倉庫」「書庫」「宝物庫」ボタンが追加される。ここからは 4

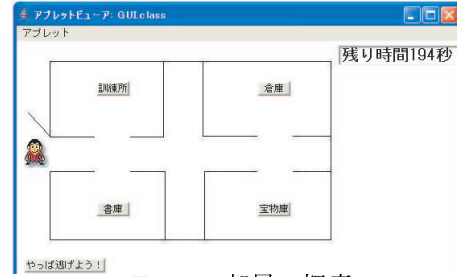


Fig. 4 部屋の探索

つの部屋を好きに何度でも出入りできるようになる。また、各部屋には、ミニゲームが用意されている。例えば倉庫では、Fig. 5 に示すゲーム画面に切り替わり、ゲームをクリアするとアイテム「鍵」が手に入る。ただし、各アイテムは一度しか入手できない。

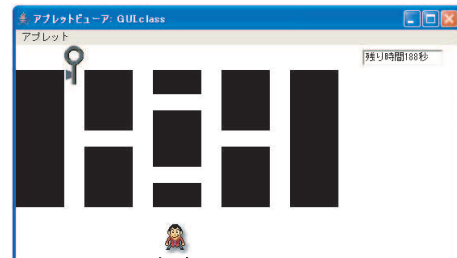


Fig. 5 倉庫におけるミニゲーム

3.3 金庫を開ける動作

「鍵」を持っている状態で、宝物庫にある金庫を調べると、金庫を開けるためのアイテム選択へと進むことができる。「鍵」は必ず選択しなければならないが、「手袋」等の他のアイテムを選択することで状態遷移の分岐がある。

お宝を入手することができれば作戦は成功したことになる、ENDING となる。

4 まとめ

今回、「泥棒」がアイテムを使ったり、手に入れたりすることによって、「手袋、針金、本、鍵、お宝」のアイテムのパラメータを変化させ、状態を遷移させる「泥棒シミュレーション」を作成した。本シミュレーションでは、アイテムの有無を表すパラメータを参照することにより、画像やボタンの有無、金庫を開けられるかどうかといったことを判断し、ユーザーに提示することができた。その結果、ユーザーは、提示された情報を元にキー操作を用いて自由な行動選択を行うことで状態を遷移し、シミュレーションを進めていくことが可能となった。さらにゲーム性向上のために 4 つのミニゲームを追加し、よりユーザーの関心を引くことが可能になった。

参考文献

- 1) 福島靖浩. Java パーフェクトマスター. 秀和システム, 2003.
- 2) 柏原正三, はじめての Java, 技術評論社, 2002.