

ラクロスゲームの作成

～逃げて敵から！決めろシュート！～

プログラミング演習 D グループ：大西 祥代

Sachiyo Onishi

1 はじめに

現在、日本のラクロス競技人口は約 1 万 8 千人である。これはサッカーや野球のようなメジャースポーツに比べると圧倒的に少ない。競技人口を増加させることは、日本ラクロスを強くするためだけでなく、ラクロスというスポーツの発展に必要なことである。そこで、多くの人々がラクロスを知るきっかけとなるよう、ラクロスというスポーツにゲーム的な要素を組み合わせ、使う人が楽しめるプログラムの作成を目指した。

2 ラクロスゲーム

ゲームは 2 つのステージに分かれている。1 つ目はディフェンスに会わずにゴールまで行く STAGE1 のゲームである。2 つ目はシュートをする STAGE2 のゲームである。ゲームの流れは、STAGE1 をクリアすると STAGE2 に進み、2 つの STAGE をクリアすればゲームクリアとなりコンピュータに勝つことができる。

2.1 プログラムの概要

2.1.1 STAGE1 のプログラム

ユーザはフィールド上を、スタート位置から縦、横、左、右、および斜めに 1 マスずつ動くことができる。移動したいマス目をマウスでクリックするとユーザのイラストが移動する。ユーザが移動できる範囲にマウスが移動すると、マス目がオレンジ色で表示される。これによりユーザが次に動くマス目を選択しやすくしている。敵の数はユーザが設定することができ、それらはランダムな位置に配置される。敵の動作はユーザと同じように 1 マスずつランダムに移動する。敵に出会うことなくゴールに辿り着くことができれば、クリア画面が表示され、STAGE1 がクリアとなる。また、ゴールするまでに敵と同じマス目に重なってしまうとゲームオーバーとなる。

2.1.2 STAGE2 のプログラム

STAGE2 では、ユーザはシューターとなりゴールにシュートを打つ。ゴールの前にはコンピュータのキーパーが配置されており、シュートを防ぐ。ゴールは、Fig1 のように 9 つのエリアに分けられている。ユーザはシュートの打ちたい場所の番号を 1～9 の中から選択する。選択したエリアにボールが飛んで行き、キーパーにセーブされることなくシュートが入ると、ユーザはポイントを

得ることができる。得られるポイント値は Fig1 のエリアごとに書かれている括弧内のものである。シュートは 5 本打つことができ、5 ポイント獲得した時点で STAGE2 クリアとなる。

1 (5P)	2 (2P)	3 (3P)
4 (2P)	5 (1P)	6 (2P)
7 (4P)	8 (3P)	9 (1P)

Fig. 1 シュートのエリア番号と獲得ポイント

2.2 オートマトン

上記のプログラムについて、Fig2 にオートマトンを示す。

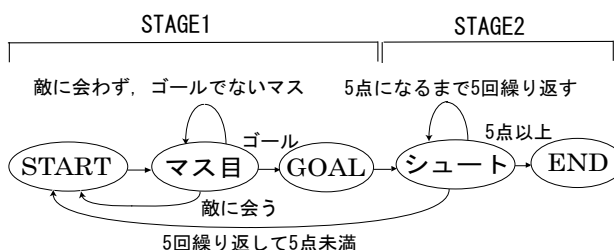


Fig. 2 ラクロスゲームのオートマトン

START から GOAL までを STAGE1 とし、シュートから END を STAGE2 とする。

1. START
状態 START はゲームの初期状態を表す。
2. マス目
ユーザが動くという状態を表す。敵に会えば状態 START に戻り、マス目が GOAL であれば状態 GOAL に遷移する。それ以外の場合には、状態 マス目を繰り返す。
3. GOAL
STAGE1 をクリアした状態を表す。
4. シュート
シュートを打つという状態を表す。シュートより得られたポイントが 5 点なら、状態 END に遷移する。5 点未満なら、シュート状態を 5 回まで繰り返す。

5回繰り返して5点未満であれば、シュートゲームに負け、状態 START に戻る。

シュートが入るか入らないかは、2つの方法で決定する。まず、選択したエリア内にボールが飛んでいくか、外れるかを乱数によって決定する。乱数はユーザがエリアを選択すると発生し、これがエリアごとに決められている値 border より大きければ、選んだエリアにボールが動く。その値よりも乱数が小さければ、ボールはゴールの枠外に飛んで行きシュートは入らない。

次に、コンピュータのキーパーによるセーブで、シュートが入るかを決定する。キーパーによるセーブは、ボールがキーパーと衝突することにより起こる。ボールがキーパーに衝突せずに選択したエリアに入れば、シュートが決まる。

エリアごとの border と、キーパーにセーブされなかった場合に得られるポイントを Table1 に示す。Table1 の r は乱数を示す。

例えば、ユーザがエリア5を選択すると、乱数が発生する。仮に 0.5 という値を得れば、 $r=0.5$ となる。Table1 より、エリア5では、 $r \geq 0.2$ でシュートボールが選択したエリア内に飛んで行き、キーパーに当たらなければシュートが入り、1ポイント獲得することができる。

Table 1 乱数による獲得ポイント

エリア番号	乱数 $r \geq \text{border}$	ポイント
1	$r \geq 0.9$	5p
2	$r \geq 0.5$	2p
3	$r \geq 0.6$	3p
4	$r \geq 0.6$	2p
5	$r \geq 0.2$	1p
6	$r \geq 0.5$	2p
7	$r \geq 0.8$	4p
8	$r \geq 0.6$	3p
9	$r \geq 0.8$	1p

3 実行結果

3.1 STAGE1 の実行結果

Fig3 は、ユーザが GOAL を目指してマス目を進んでいる様子である。丸で囲んでいるイラストがユーザを表し、バイキンマンが敵を表している。ユーザの8方向のマス目にマウスを移動させると、Fig3 のように色が変化し、移動できる範囲を示す。GOAL という文字が書かれているマス目にユーザが到着するとクリア画面が現れ、STAGE2 に進む。

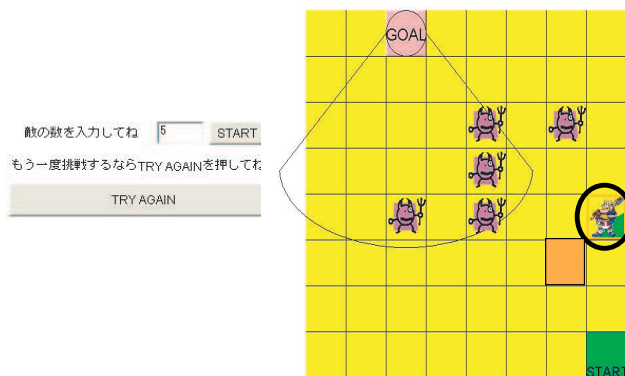


Fig. 3 STAGE1 の GUI

3.2 STAGE2 の実行結果

Fig4 は、STAGE2 の表示画面である。右側の画面で、1～9のボタンを用いて打つ場所を決定し、残りのシュート本数と、現在の得点を表示している。左側の画面にシュートが決まる、もしくは外れる様子を表示する。Fig4 では、シュートが決まった様子を表示している。ここでは残りのシュート本数が2本で、獲得した総得点が3点と表示されている。



Fig. 4 STAGE2 の GUI

4 まとめ

ラクロスをゲーム上にシミュレートできるように、ゴールまでボールを運ぶゲームとシュートを行うゲームのプログラムを作成した。プログラムを実行し、STAGE1 と STAGE2 が正常に動作することを確認した。

参考文献

- 1) Steven Holzner, JAVA プログラミング BlackBook 2nd Edition, 株式会社インプレス, 2002