

プログラミング概論

第12回 2023年12月6日

App Inventorによる

Androidアプリ開発の実践

(6) 物理シミュレーション3

今回の授業内容

- 的当てゲームの発展
 - 地面での跳ね返り
 - 射出速度のコントロールを改良する
 - 的当てゲームをアレンジする
- 完成したらレポートを書く

前回同様に下準備として、
前回のプロジェクトを別名で保存して
から作業を始めよう

的当てゲームの発展

地面での跳ね返り

地面ではずむボールの動きをイメージしてほしい。よくはずむボールでも地面に衝突するたびに跳ね返ってくる距離は短くなるのが想像できると思う。

現象として捉えると、X軸方向の動きは一定、Y軸方向の動きは、地面との衝突によって、速度の符号が逆になり（=下向きの速度が上向きになる）、かつその大きさがe倍（ $e < 1$ ）になっている（=速度が小さくなる、減衰する）ことに相当する。

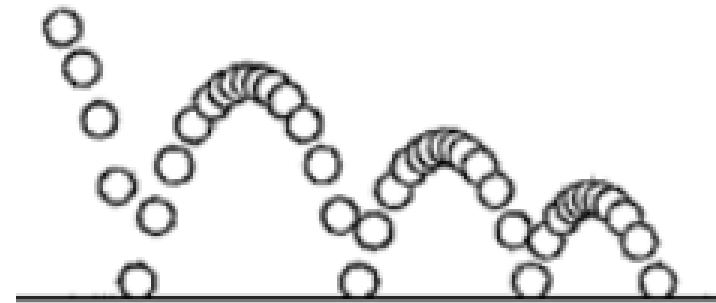
```
when Ball1 .EdgeReached
  edge
do set Ball1 .Visible to false
```



```
initialize global e to 0.8

when Ball1 .EdgeReached
  edge
do if get edge = -1
  then set global vy to neg get global e × get global vy
  else set Ball1 .Visible to false
```

北 : 1
南 : -1
東 : 3
西 : -3



球が画面端に当たったとき
南なら
y軸方向の速度を-e倍する

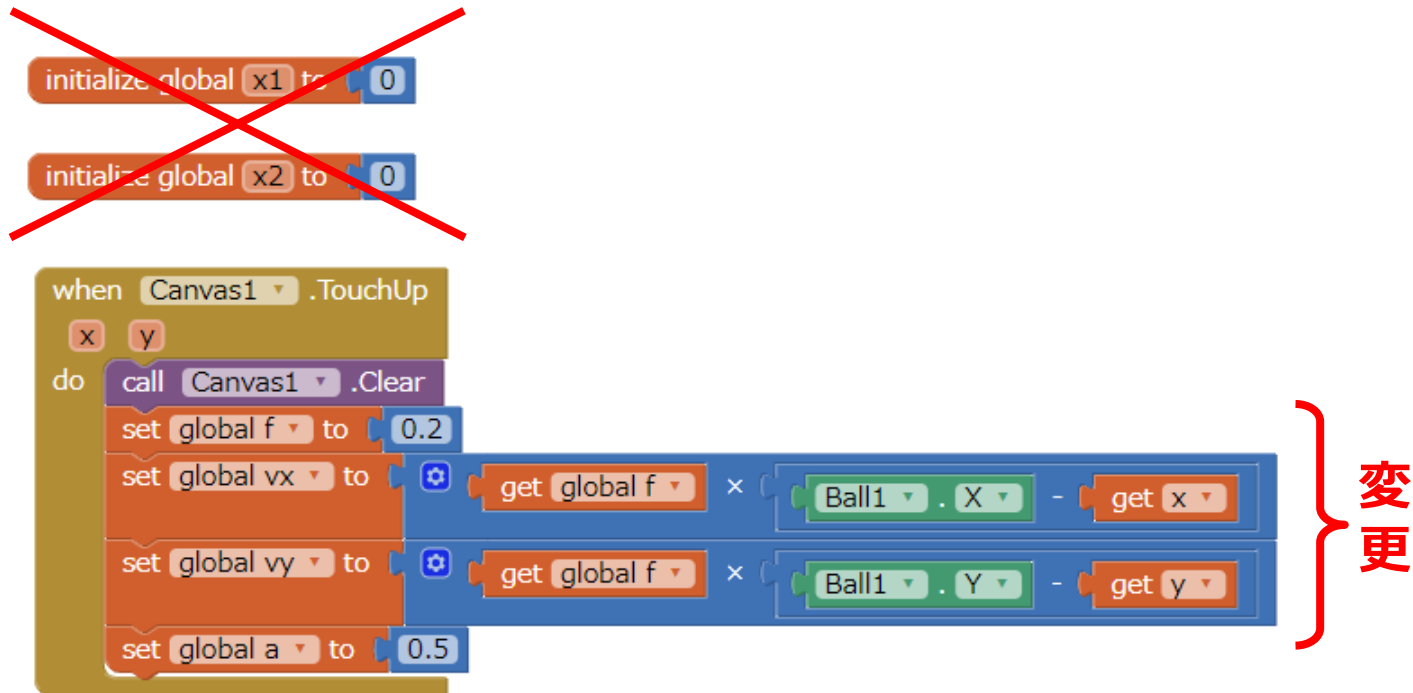
それ以外なら
見えなくする

やってみよう

- 球が的に当たった時も速度が減衰するように変更してみよう
- バウンド回数をカウントして表示するようになしてみよう

射出速度のコントロールを改良する

前回は水平方向の射出速度をタップした場所のx座標と離れた場所のx座標から求めた。今回は、画面から指を離すと球の初期位置の座標と離れた場所の座標の差に応じた速度で射出されるように変更する。また、x軸成分だけでなくy軸成分も射出できるように改良する。



The image displays Scratch code blocks. The top two blocks, 'initialize global x1 to 0' and 'initialize global x2 to 0', are crossed out with a large red 'X'. Below them is a 'when Canvas1 TouchUp' event block with a 'do' loop containing the following blocks:


- call Canvas1 Clear
- set global f to 0.2
- set global vx to $\text{get global f} \times (\text{Ball1 X} - \text{get x})$
- set global vy to $\text{get global f} \times (\text{Ball1 Y} - \text{get y})$
- set global a to 0.5

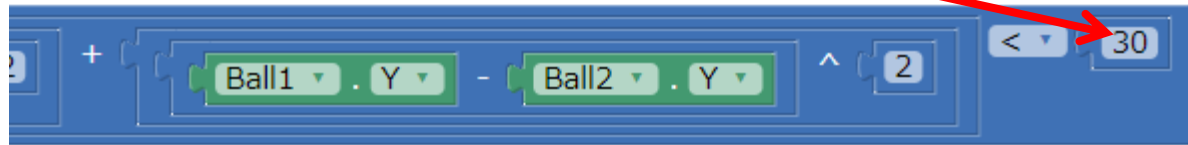
A red bracket on the right side of the code blocks is labeled with the Japanese word '変更' (change), indicating the modification to the velocity calculations.

的当てゲームをアレンジする

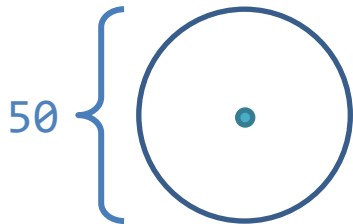
- Resetするたびに的のx座標が変わる
- Resetするたびに的の大きさが変わる
- 的が常に移動する
- 通ってはいけないエリアを設ける
- 障害物を配置する
- オリジナルの工夫も考えてみよう

ヒント集

- 的の半径 
- 球や的の半径を変えるならここも変える必要がある

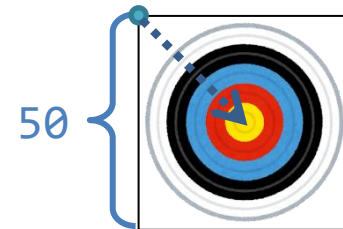


- 球や的を画像にする場合はImageSpriteを使う。ただし座標の基準が中心ではなく左上になるので配置や衝突判定をずらす必要がある。



Ball2 . X

Ball2 . Y



ImageSprite1 . X + 25

ImageSprite1 . Y + 25

完成したらレポートを書く

課題レポートについて

- レポートの書き方
 - Microsoft Wordで作成する
 - レポートの冒頭に、**科目名・タイトル・所属・学生番号・氏名・提出年月日**を入れる
 - 決められた構成で書く
 - サンプルをよく見てまねて書く
 - 提出方法・期限を守る
 - 困ったことがあったら**期限前に**担当教員に相談する → 1G2G: akiyo@mail.tohoku-gakuin.ac.jp
3G4G: hiro_m@mail.tohoku-gakuin.ac.jp

課題レポートについて

- レポートの構成
 1. アプリの概要
 - 特徴・機能など詳細に書く。
 2. デザイナ画面
 - 各コンポーネントの説明も入れる。
 3. ブロックエディタ画面
 - 各変数の説明も入れる。
 4. 実行画面・操作方法
 5. 工夫した点
 - 「なし」はダメ。何か工夫すること。

第2回課題レポートについて

- タイトル
 - 的当てゲームの制作
- 内容
 - 制作した的当てゲームについてレポートを作成する
- 提出方法
 - 前回と同じ
- 提出期限
 - 12月19日（火） 23:59

レポート作成上の注意

- 「コンポーネントの説明」「変数の説明」「操作説明」を書き漏らさないよう注意すること。
- ブロックエディタ画面のスクリーンショットは**すべてのブロックが読めるように貼り付ける**こと。
- 「工夫した点」を必ず挙げること。背景画像の追加などデザイナー画面でおこなった工夫を含めても良いが、最低でも一つは「プログラム上の工夫（ブロックエディタ上の工夫）」があること。
- 「工夫した点」については、（プレイしなくても読んだだけで）意味が伝わるように特に丁寧に説明を書くこと。
- レポートは感想文ではないので感想を書いてはいけない。
 - 「～できてよかった」「大変だった」「難しかった」など