

プログラミング概論

第11回 2023年11月29日

App Inventorによる

Androidアプリ開発の実践

(5) 物理シミュレーション2

今回の授業内容

今回は**放物運動**のシミュレーションを行うことを目的とする。
放物運動とは**空中に投げられた物体の運動**のことである。
重力により、物体には鉛直下向きに行こうとする力が常にかかっている。

空気抵抗などの重力以外の力を無視し、投げられたものの本質的な動きだけをシミュレートする。

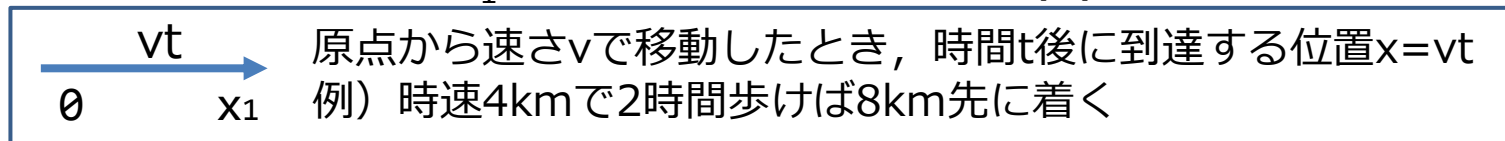
- 等速直線運動
- 等加速度運動（水平方向）
- 落体の運動（鉛直方向の等加速度運動）
- 放物運動
- 的当てゲーム（放物運動版）を作る

放物運動とは

等速直線運動

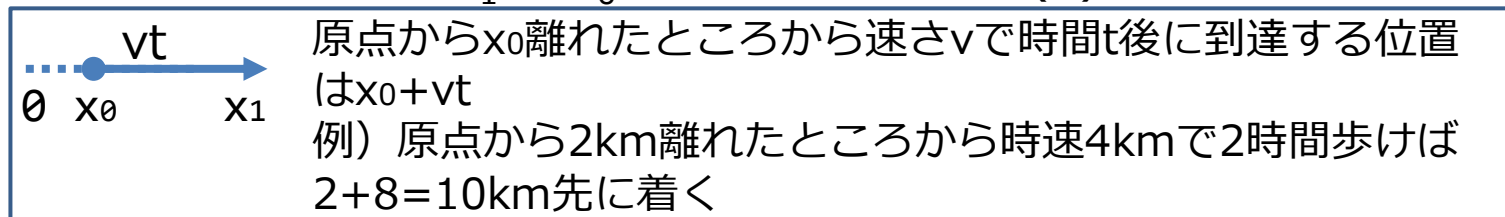
1次元の直線に沿った動きを考える。物体の運動とは物体の位置が時間的に変化することをいう。今、原点からx軸方向に一定の速さvで移動しているとき（等速直線運動）、時刻t後に到達する位置 x_1 は以下の式で表すことができる。

$$x_1 = vt \quad (1)$$



原点ではなく、位置 x_0 からの移動を考えるなら、次の式のようになる。

$$x_1 = x_0 + vt \quad (2)$$



この式(2)を変形すると $x_1 - x_0 = vt$ となる。これが非常に短い時間に起きていることであると考えてみよう。非常に短い時間（微小時間） Δt の間に、非常に短い距離（微小距離） Δx だけ移動することを式で表すと

$$x - x_0 = \Delta x = v\Delta t \quad (3)$$

この式の意味を改めて書き直してみると「速さvで移動している物体は微小時間 Δt の間に Δx だけ移動する」ということになる。

等速直線運動

等速直線運動のサンプルプログラムを以下に示す。やっていることは前回のプログラムと同様であるが、前ページの運動論に基づいた記述にする。ポイントは、

微小時間 dt の間の位置の変化量 dx を速さ v_x を用いて



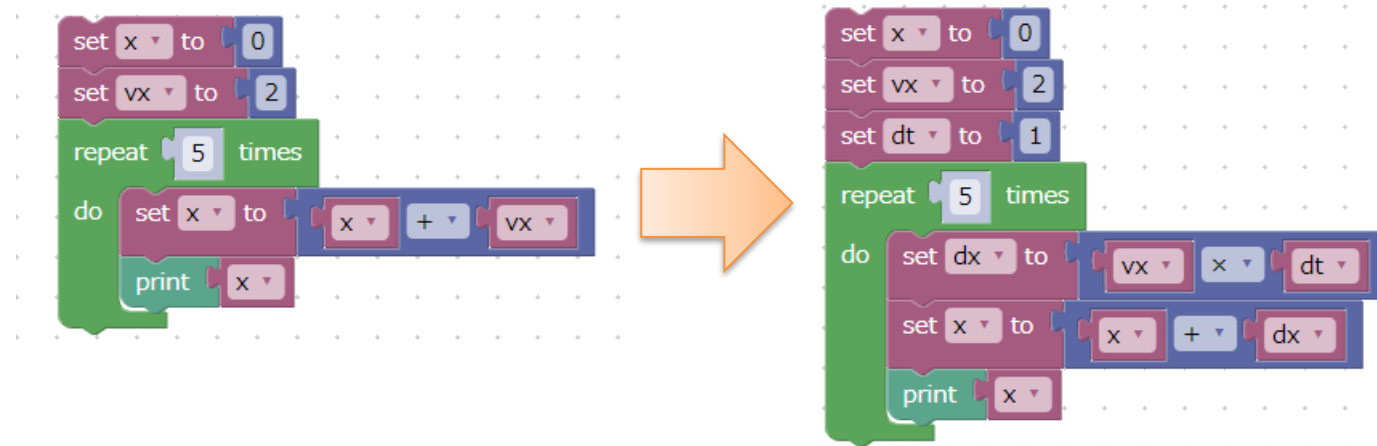
Scratch code block: set dx to vx * x * dt

としていること、並びに、この位置の変化量 dx を用いて



Scratch code block: set x to x + dx

によって位置情報を更新しているところである。



等加速度運動（水平方向）

加速度とは「単位時間あたりの速度の変化量」を指す。自動車でいえば、アクセルの踏み込みに相当する。秒速10mの速さ（※1秒あたり10m進む速さ。1時間=3600秒に換算すると36000m=36Km,つまり時速36Kmのこと）で走っていたところで、アクセルをぐっと踏無ことで秒速20m（時速72Km）になったとしよう。もし1秒間の間に10m/sの速さから20m/sの速さに変化したのであれば、加速度は $(20-10)/1=10$ （単位は $(\text{m/s})/\text{s}=\text{m/s}^2$ ），もし10秒間で20m/sの速さに達したのであれば、加速度は $(20-10)/10=1\text{m/s}^2$ ということになる。これを数式で表すと次のようになる。

$$\text{加速度}a = (v_2 - v_1) / t \quad (4)$$

この式を変形すると $v_2 - v_1 = at$ となる。先と同様に、微小時間 Δt の間に生じた微小な速さの変化 Δv として式を書き直すと

$$v_2 - v_1 = \Delta v = a\Delta t \quad (5)$$

となる。

等加速度運動をプログラムとして表現するために次の段を踏むこととする。

- ① 加速度から速さの変化量を求める
- ② 速さの変化量に基づいて、速さの値を更新する
- ③ 更新された速さに基づいて移動距離の変化量を求める
- ④ 移動距離の変化量に基づいて、位置の値を更新する

等加速度運動（水平方向）

①表示される値を考えてみよう

②aが-1のとき表示される値を考えてみよう

```
set x to 50
set vx to 0
set dt to 1
set a to 1
repeat 5 times
do
  set dv to a x dt
  set vx to vx + dv
  set dx to vx x dt
  set x to x + dx
print x
```

①

/	/	/	/
---	---	---	---

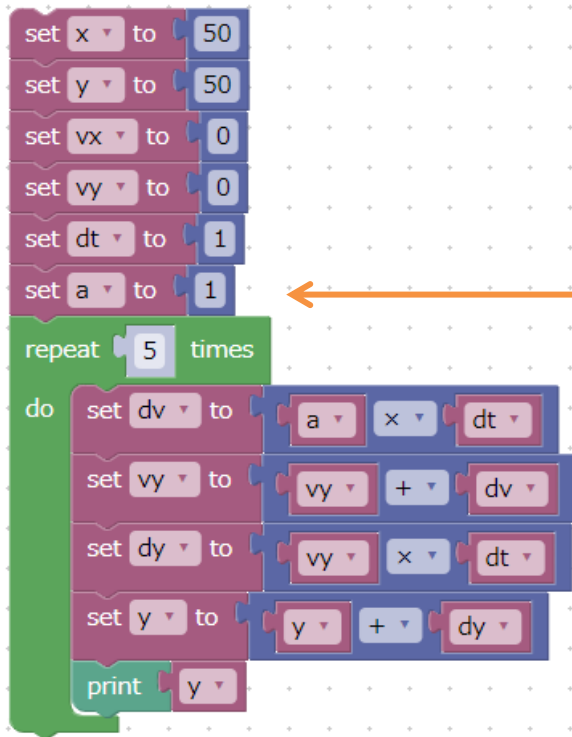
②

/	/	/	/
---	---	---	---

加速度は負の値になっても良い。一定の速さで動いていけば、負の加速度は減速の効果をもたらす。速さが0になっても負の加速度が与えられたままであれば、そこから逆向きの加速度運動になる。

落体の運動（鉛直方向の等加速度運動）

重力がかかっているところでの動きをシミュレートする。モノを落とすと落下速度は大きくなっていくが、実は加速度は一定である。重力によってもたらされる加速度のことを重力加速度といい、その値は約 9.8m/s^2 である。1秒ごとに移動速度が 9.8m/s ずつ増えていく、という量である。車になぞらえれば、止まっている状態から2秒間で時速約 70km に達するような量ということもできる。



```
set x to 50
set y to 50
set vx to 0
set vy to 0
set dt to 1
set a to 1
repeat 5 times
do
  set dv to a * dt
  set vy to vy + dv
  set dy to vy * dt
  set y to y + dy
print y
```

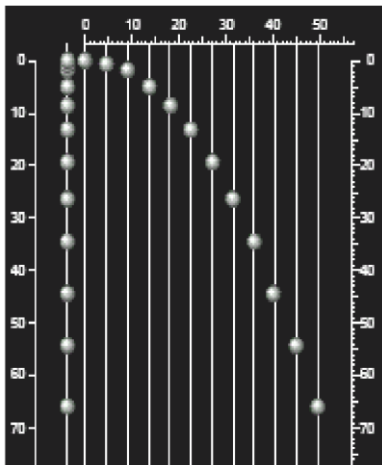
なお、このシミュレーションでは、プログラムの分かりやすさと可視化を重視し、現象のリアリティについては厳密性を求めないこととする。つまり、どれくらいの長さを 1m と見立てるか？といった換算は行わない。よって、重力加速度も 9.8 という値を用いず、見易さを重視し、小さな値を用いることとする。

y軸方向の初期速度として負の値（例えば $vy=-10$ ）にするとどのような動きを示すか考えてみよう

放物運動

冒頭でも述べた通り，放物運動とは空中に投げられた物体の運動のことを指す．実際に投げた時の様子を図1に示す．図中（左）は単なる落体の運動について，図中（右）は水平方向に球を投げ出した時の運動について，一定時間ごとの位置を示したものである．

この図から分かるのは，**水平方向には等速運動**，**鉛直方向には「落体の法則」に従った等加速度運動**になっていることである．よって、モノを投げる状態をシミュレートするためには運動を「重力がかかる鉛直方向（Y軸）」と「それに直交する水平方向（X軸）」の成分に分解し，水平方向と鉛直方向のそれぞれで計算を独立に行って，それを単純に足し合わせればよいことになる．



<https://ocw.hokudai.ac.jp/wp-content/uploads/2016/01/BasicPhysicsI-2007-slide-all.pdf>

```
set x to 50
set y to 50
set vx to 2
set vy to 0
set dt to 1
set a to 1

repeat 5 times
do
  set dx to vx * dt
  set x to x + dx
  set dv to a * dt
  set vy to vy + dv
  set dy to vy * dt
  set y to y + dy

print create text with x , y
```



的当てゲームを作る



前回のプロジェクトを別名で保存する

The screenshot shows the MIT App Inventor web interface. At the top, a blue banner contains the title "前回のプロジェクトを別名で保存する". Below it, the browser address bar shows "ai2.appinventor.mit.edu/#4783042956492800". The main navigation bar includes "Projects", "Connect", "Build", "Settings", "Help", "My Projects", "View Trash", "Guide", "Report an Issue", "English", and "akiyolab5@gmail.com". A red arrow points to the "Projects" menu, which is open, showing a list of actions. A second red arrow points to the "Save project as ..." option in this menu. The interface also shows a "Matoate" sidebar with various component palettes (User Interface, Layout, Media, Drawing and Animation, Maps, Charts, Data Science, Sensors) and a central workspace displaying a mobile app preview with a red circle on the screen. The right side features an "All Components" panel and a "Properties" panel for the selected "Screen1 (Form)".

MIT APP INVENTOR

Projects Connect Build Settings Help My Projects View Trash Guide Report an Issue English akiyolab5@gmail.com

Matoate

Project Properties Publish to Gallery Designer Blocks

My projects

- Start new project
- Import project (.aia) from my computer ...
- Import project (.aia) from a repository ...
- Move To Trash
- Save project
- Save project as ...
- Checkpoint
- Project Properties...
- Export selected project (.aia) to my computer
- Export all projects
- Import keystore
- Export keystore
- Delete keystore

Screen1

- Canvas1
 - Ball1
 - Ball2
- HorizontalArrangement1
 - Button1
 - Label1
 - Label2
 - Label3
- Clock1

Properties

Screen1 (Form)

Appearance

AboutScreen

AlignHorizontal

AlignVertical

BackgroundColor

BackgroundImage

BigDefaultText

CloseScreenAnimation

HighContrast

Privacy Policy and Terms of Use

前回のプロジェクトを別名で保存する

The screenshot shows the MIT App Inventor web interface. At the top, a blue banner contains the title "前回のプロジェクトを別名で保存する". Below it, the browser address bar shows "ai2.appinventor.mit.edu/#4783042956492800". The MIT App Inventor logo and navigation menu are visible. A notification bar indicates "Saved project at Nov 28, 2023, 11:18:15 PM". The main workspace is titled "Matoate" and includes a "Save As" dialog box. The dialog box has a "New name:" field containing "Matoate2". A blue arrow points to this field with the text "今回は「Matoate2」". A red arrow points to the "OK" button. The background interface includes a "Palette" with categories like "User Interface", "Layout", "Media", etc., and a "Properties" panel on the right.

アプリのタイトルを変える

The screenshot shows the MIT App Inventor web interface. At the top, a blue banner contains the title "アプリのタイトルを変える". Below it, the browser address bar shows the URL "ai2.appinventor.mit.edu/#4783042956492800". The interface includes a navigation menu with options like "Projects", "Connect", "Build", "Settings", "Help", "My Projects", "View Trash", "Guide", and "Report an Issue". On the left, a component palette lists various widgets such as "Spinner", "Switch", "TextBox", "TimePicker", and "WebView". The central workspace displays a mobile app preview with a white screen and a black navigation bar. To the right of the preview, there are "Rename" and "Delete" buttons. Below these, a "Media" section contains an "Upload File ..." button. On the far right, a settings panel lists various application properties. The "Title" property is highlighted with a red box and contains the text "的当てゲーム". A blue arrow points from the Japanese text "前回とは異なる方が混乱しない" to this "Title" property.

MIT APP INVENTOR

Projects Connect Build Settings Help My Projects View Trash Guide Report an Issue English akiyolab5@gmail.com

Spinner Switch TextBox TimePicker WebView

Layout Media Drawing and Animation Maps Charts Data Science Sensors Social Storage Connectivity LEGO® MINDSTORMS® Experimental

CloseScreenAnimation Default HighContrast OpenScreenAnimation Default PrimaryColor Default PrimaryColorDark Default ScreenOrientation Unspecified Scrollable ShowListsAsJson ShowStatusBar Title 的当てゲーム TitleVisible Application

Rename Delete

Media Upload File ...

Privacy Policy and Terms of Use

前回とは異なる方が混乱しない ---->

変数の追加と初期化

initialize global a to 0

加速度

initialize global dt to 1

微小時間

initialize global dv to 0

initialize global dx to 0

initialize global dy to 0

initialize global f to 0

initialize global x1 to 0

タップした位置 (x座標)

initialize global x2 to 0

指を離した位置 (x座標)

```
when Button1 .Click
do
  set global vx to 0
  set global vy to 0
  set Ball1 . X to 50
  set Ball1 . Y to 50
  set Ball1 . Visible to true
  set global total to [get global total + 1]
  set Label1 . Text to [join [get global score "/ " get global total]]
  set global a to 0
```

追加

球を飛ばす処理

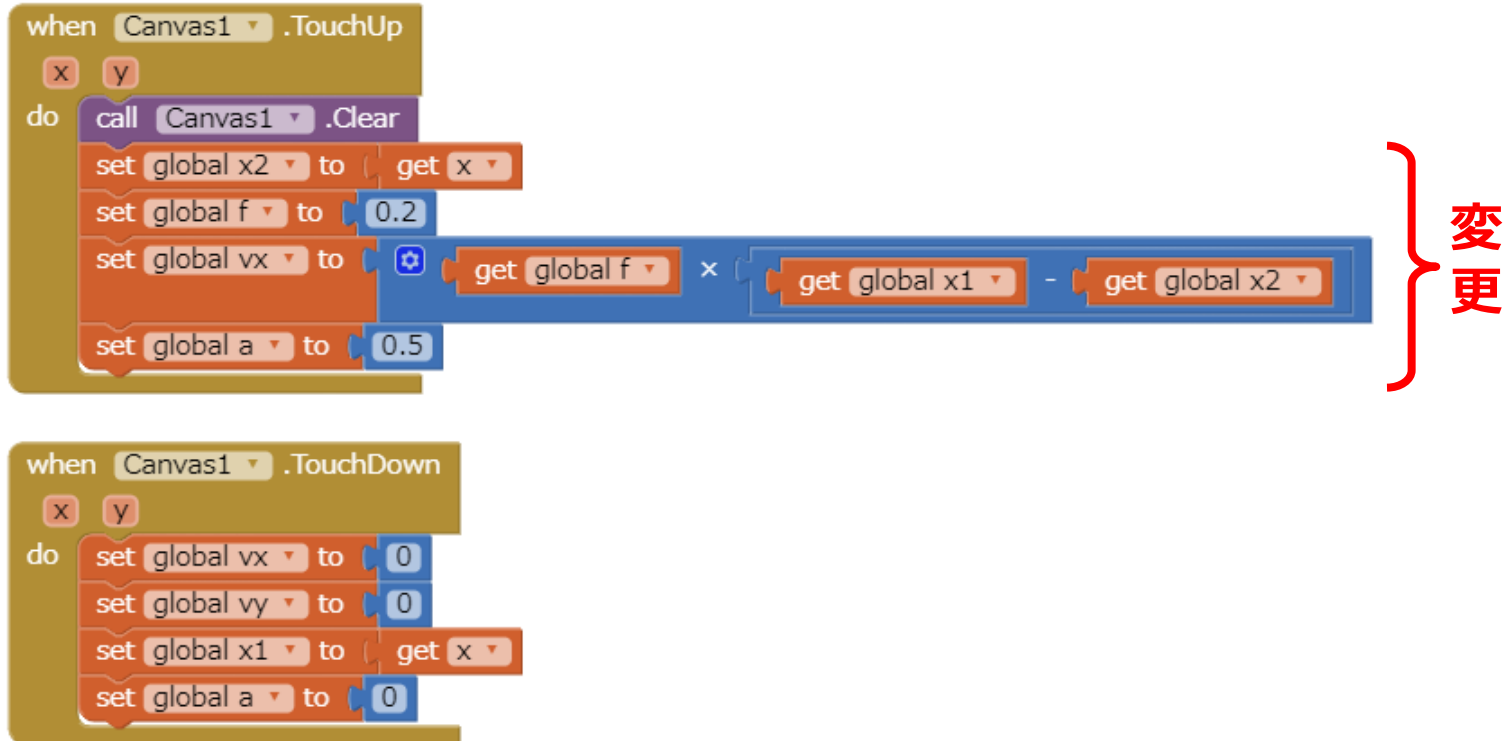
The image shows a Scratch script for moving a ball. The script is as follows:

```
when Clock1 .Timer do
  set global dx to [ ]
  set Ball1 . X to [ ]
  set global dv to [ ]
  set global vy to [ ]
  set global dy to [ ]
  set Ball1 . Y to [ ]
  if (square root of ((Ball1 . X - Ball2 . X)^2 + (Ball1 . Y - Ball2 . Y)^2) < 30) then
    set global vy to (get global vy) * (-1)
    set global vx to (get global vx) * (-1)
    set global score to (get global score) + 1
  set Label2 . Text to Ball1 . X
  set Label3 . Text to Ball1 . Y
```

The following table summarizes the actions in the code blocks:

Code Block	Japanese Description
set global dx to	水平方向の移動量を計算
set Ball1 . X to	水平方向の位置情報を更新する
set global dv to	鉛直方向の速さを計算
set global vy to	鉛直方向の位置の計算
set global dy to	鉛直方向の位置情報を更新する
set Ball1 . Y to	鉛直方向の位置情報を更新する
if (square root of ((Ball1 . X - Ball2 . X)^2 + (Ball1 . Y - Ball2 . Y)^2) < 30)	変更
set global vy to (get global vy) * (-1)	鉛直方向の速さを反転させる
set global vx to (get global vx) * (-1)	水平方向の速さを反転させる
set global score to (get global score) + 1	スコアを1増やす
set Label2 . Text to Ball1 . X	Label2のテキストをBall1のX座標に更新する
set Label3 . Text to Ball1 . Y	Label3のテキストをBall1のY座標に更新する

タップしたときと離れたときの処理



```
when Canvas1 .TouchUp
  x y
  do
    call Canvas1 .Clear
    set global x2 to get x
    set global f to 0.2
    set global vx to get global f * (get global x1 - get global x2)
    set global a to 0.5

when Canvas1 .TouchDown
  x y
  do
    set global vx to 0
    set global vy to 0
    set global x1 to get x
    set global a to 0
```

変更

これ以外のブロックは変更なし

工夫してみよう

- Resetするたびに的のx座標が変わるようにしてみよう
- Resetするたびに的の大きさが変わるようにしてみよう
- 的が常に移動するようにしてみよう
- オリジナルの工夫も考えてみよう