運動シミュレーション2
ロケットの逆噴射・雨粒の落下

早大本庄　情報科　飯島　涼

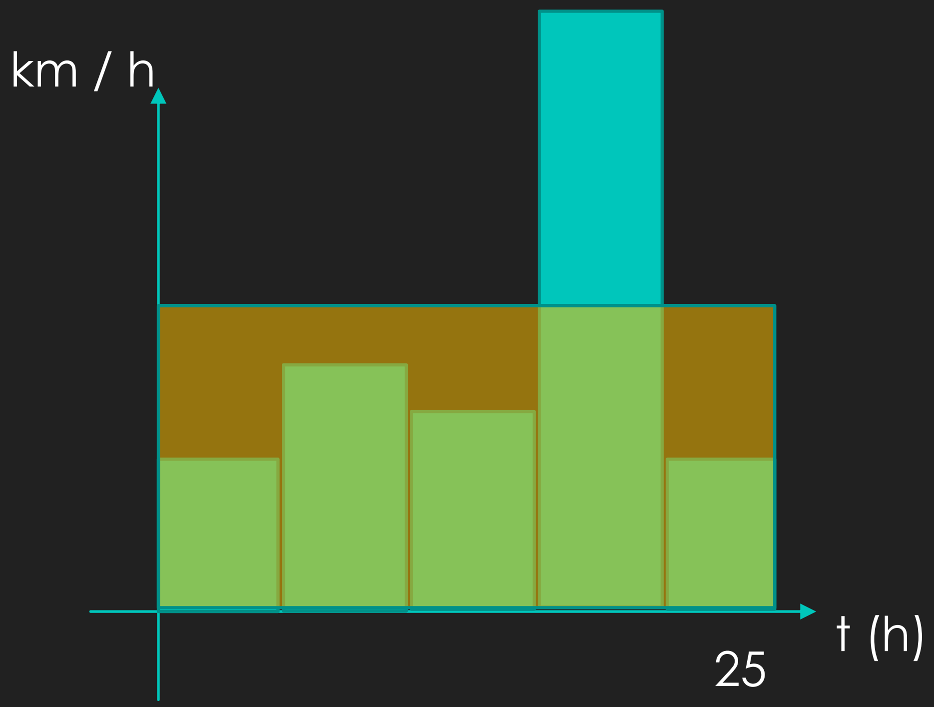
なぜプログラミングで運動?

- ゲームプログラミング
- 工学全般
- ドローン・車・航空機の自動運転の制御
  - https://www.youtube.com/watch?v=q4Ugu4iJukM
- 車・航空機のシミュレーション（次回）

https://game.watch.impress.co.jp/docs/news/1274927.html

時間の幅を狭くする

- 最初の5時間で15km　$\frac{15}{5} = 3$
- 次の5時間で　20km
- 次の5時間で　10km
- 次の5時間で　50km
- 次の5時間で　5km　$\frac{5}{5} = 1$



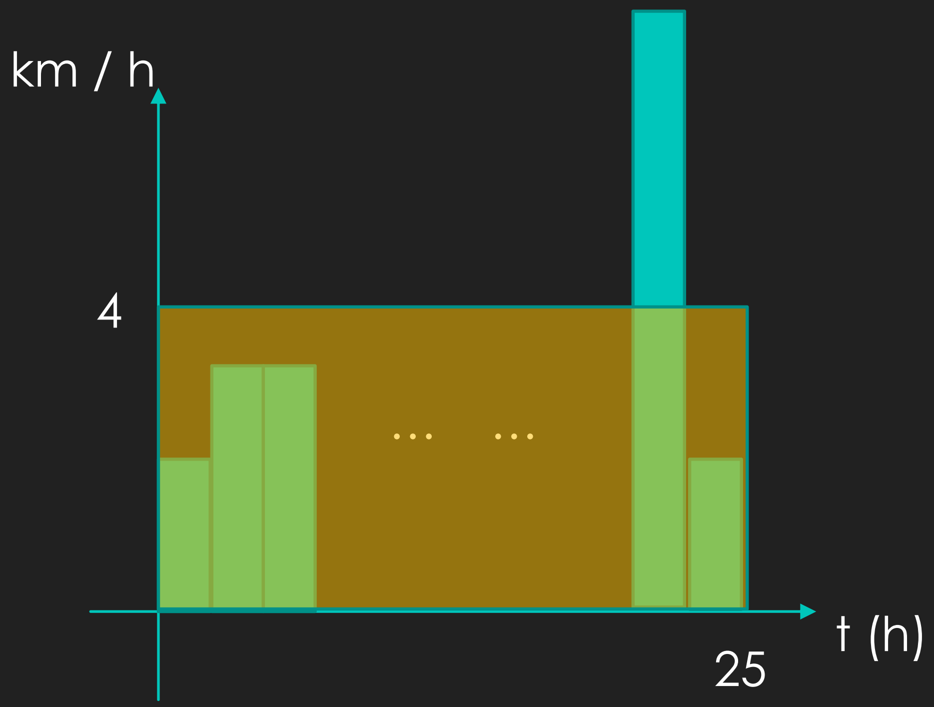
時間の幅を狭くする

- 最初の1時間で2km　$\frac{2}{1}$
- 次の1時間で　3km
- 次の1時間で　3km　$\frac{3}{1}$

…

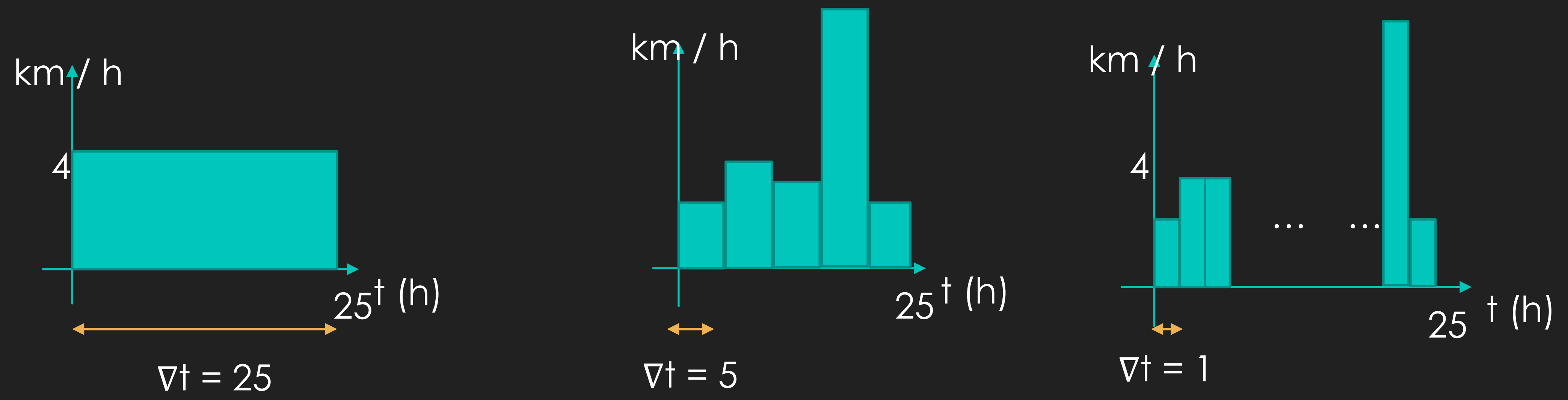
……

- 次の1時間で　20km　$\frac{20}{1}$
- 次の1時間で　2 km



Δt: 区切る時間の幅

$$\frac{100}{25} \Rightarrow \frac{15}{5} \Rightarrow \frac{3}{1}$$

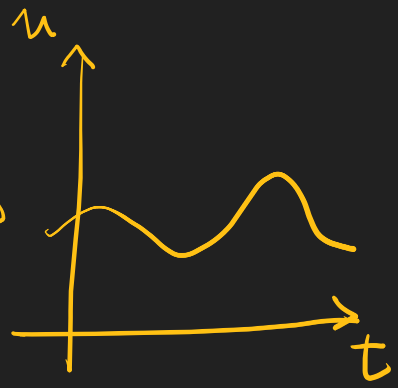


速度とは

- 区切る時間幅Δt　が小さいほど，今この瞬間の速度が正確にわかる

Q: 区切る時間幅Δtをかぎりなく小さくしていったらどうなる？

$$\frac{100}{25} \Rightarrow \frac{15}{5} \Rightarrow \frac{3}{1}$$



時刻tでの位置を，$x(t)$とすると，

$$\lim_{\Delta t \to 0} \frac{x(t+\Delta t) - x(t)}{\Delta t} = v(t)$$

瞬間速度 v (m/s)

区切る時刻Δt を，限りなく小さくしたときの速さのこと

$$v(t) = \lim_{\Delta t \to 0} \frac{x(t+\Delta t) - x(t)}{\Delta t} = x'(t)$$

公式の復習

初期位置をx_0, 初速度をv_0, 重力加速度をgとしたとき，*x(t)は,*

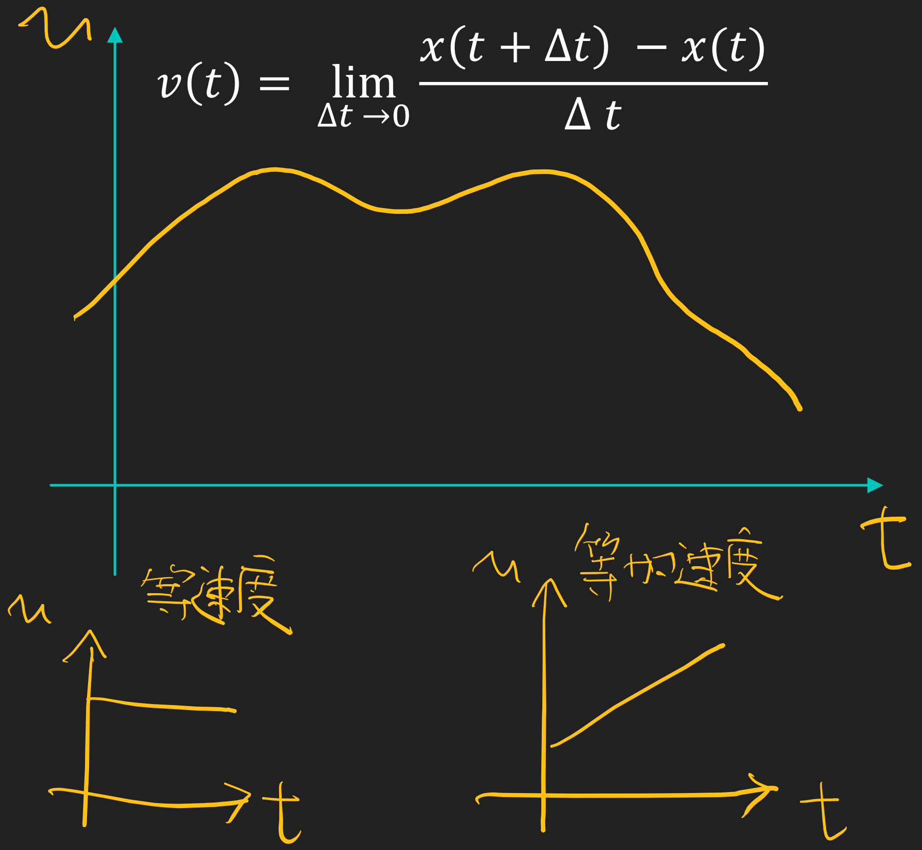
$$x(t) = \frac{1}{2}gt^2 + v_0 t + x_0$$

$$v(t) = x'(t) = gt + v_0$$

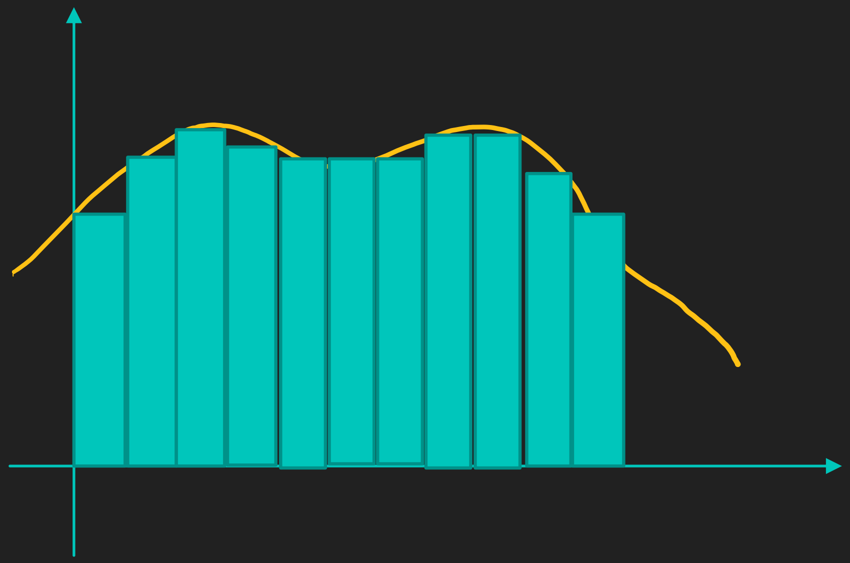
物理の理論とプログラミングの違い

- 物理（理論）

$$v(t) = \lim_{\Delta t \to 0} \frac{x(t+\Delta t) - x(t)}{\Delta t}$$



- プログラミング

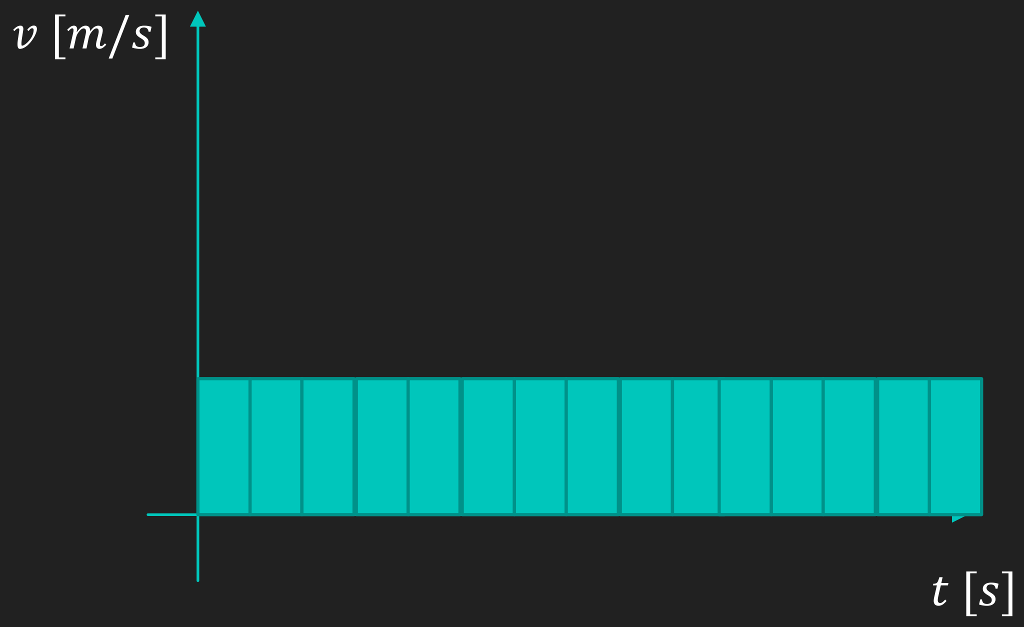


Δtをある程度の小ささで妥協して，
運動の振る舞いを観測してみる

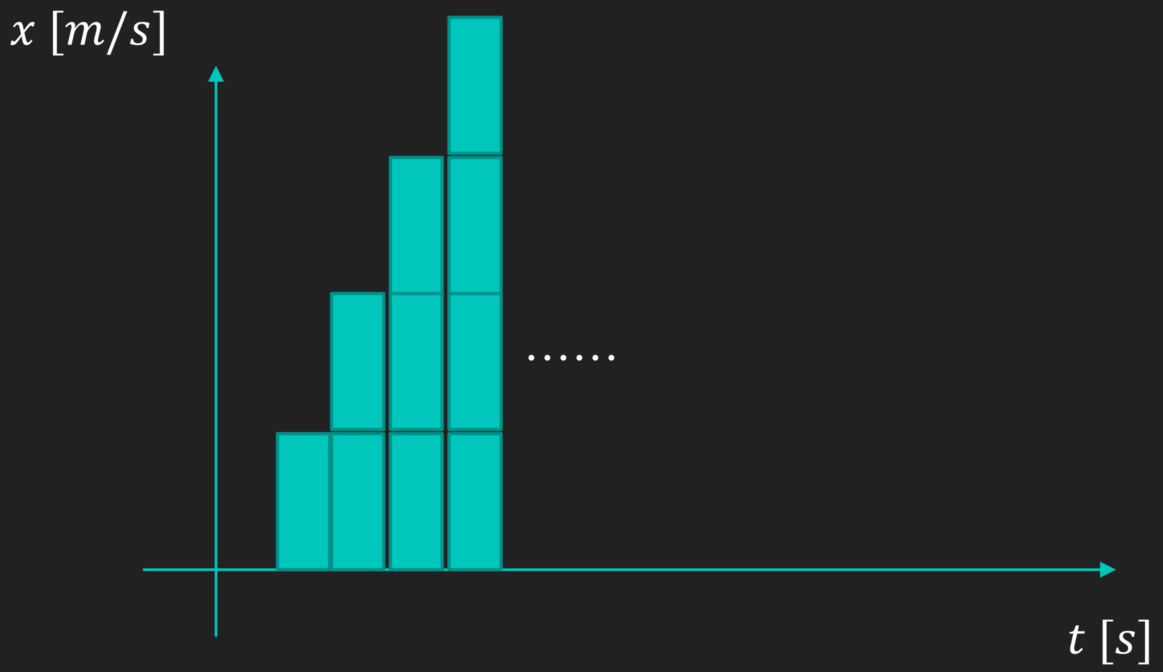
離散的なデータ

運動シミュレーションの基礎

- 等速度運動



変数 v (固定値を代入)



変数 x (時間経過ごとに位置の変化を足していく)

プログラムと運動との対応

```
v0 = 10 #速度 v0 [m/s]
x0 = 0.0 # 位置 x0 [m]

t = 0.0 # 時間 t[s]
dt = 0.01 # 区切る時間の幅Δt

TIME_END = 20

while int(t) < TIME_END:
① t += dt
②
```

0.01秒ごとの経過

①
1回分の繰り返しを， Δt 秒の時間経過とみなす

②
Δt 秒時間が経過するごとに，その位置をリストに記録していく
=> ストロボ写真

加速度 $a[m/s^2]$

- プログラミング用の定義: 単位時間あたりに，速度に加える値のこと

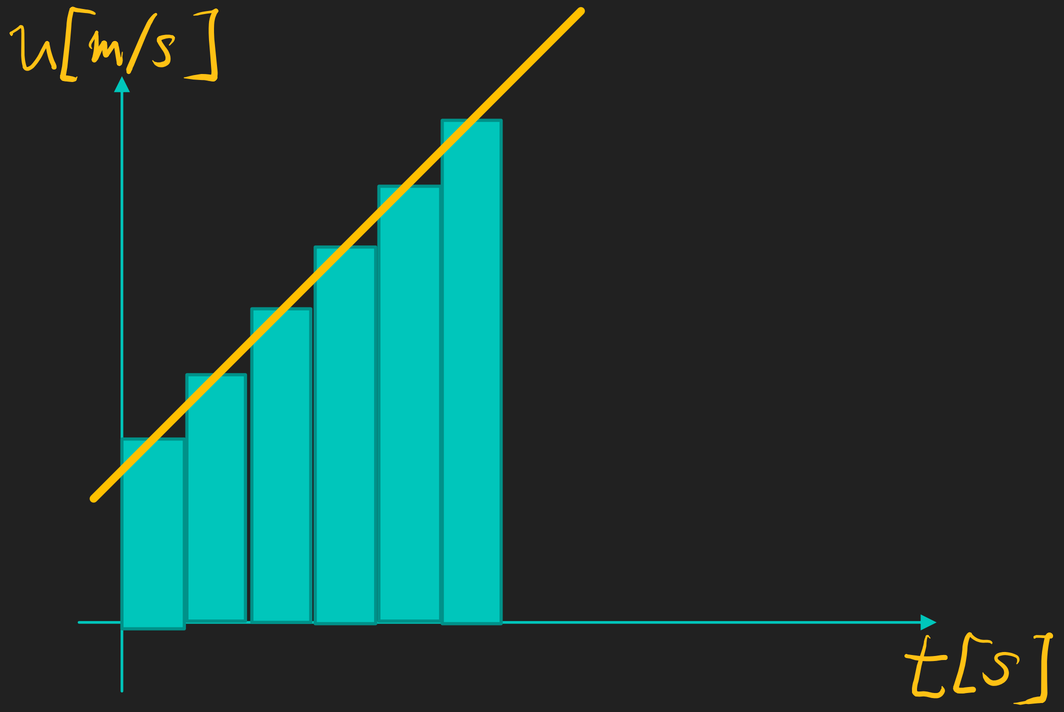
$\boldsymbol{a[m/s^2]}$:　1[s]あたり，a[m/s] を速度に加える

例)

$\mathbf{5}[\boldsymbol{m/s^2}]$:

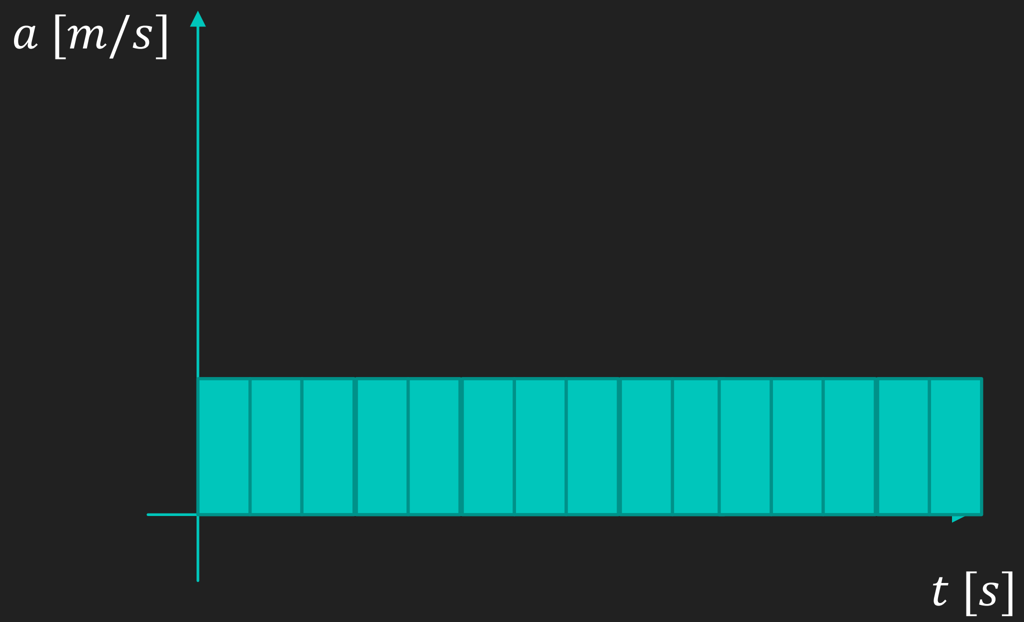
1[s]あたり，5[m/s] を速度に加える

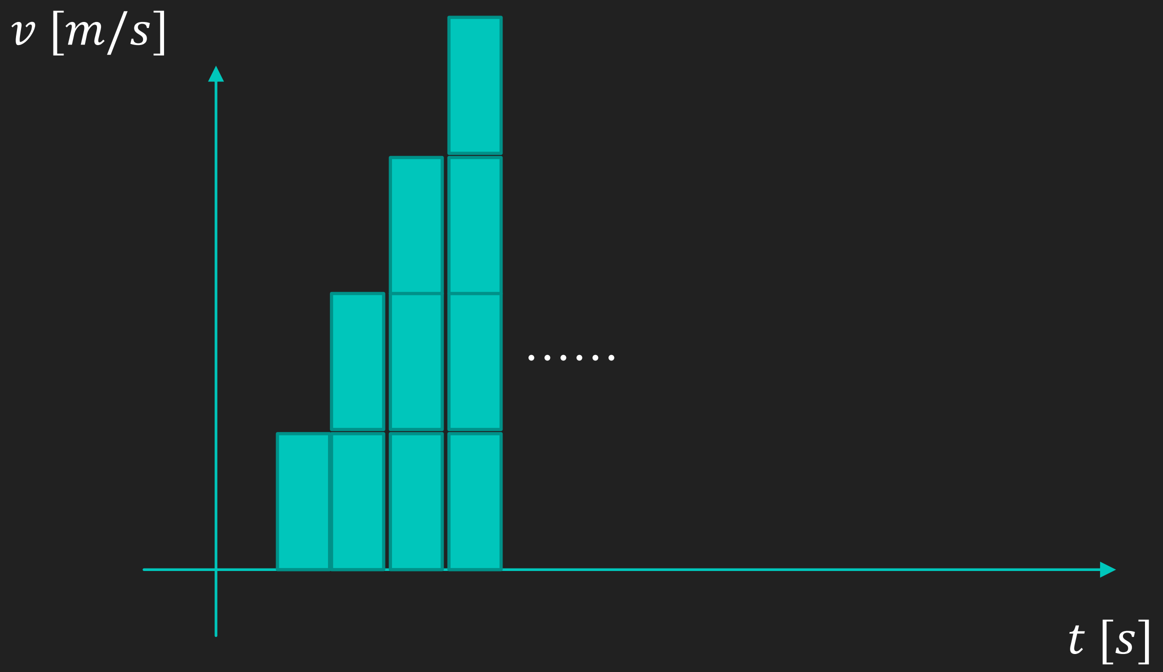
0.01[s] あたり，$0.01 \times 5[m/s]$



運動シミュレーションの基礎

- 等加速度運動





変数 a (固定値を代入)

変数 v (時間経過ごとに速度の変化を足していく)

シミュレーションまとめ

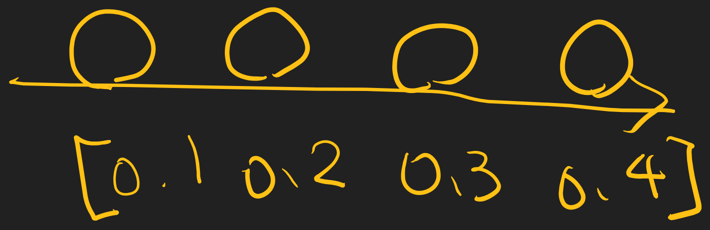
- オイラー法
  - Δtごとに時間を進めて，速度や位置を記録していくシミュレーション方法

○ 100

○ 99

○ 97

○ 92



なぜ公式をグラフ化しない?

- グラフ化するための式を手計算しなくてはいけない(本日の自由課題の例)

- そもそも式にするのが困難な運動がある
  - 運動に伴って，質量が変化する（雪だま）
  - 物体に加わるFの値が時間によって変化するもの（次回，ロケット噴射）

- 公式を知らなくても，どのようにふるまうのか知らなくても初期値がわかれば試せる．（本日）

本日の内容

- 小数の誤差についての補足（エラーの原因その3）

- 公式で表しにくい運動をシミュレーションする
  - ロケットの逆噴射
  - 雨粒の落下

参考資料

- 大重美幸、詳細！Python 3 入門ノート、ソーテック社、2017
- 小高知宏、Pythonによる数値計算とシミュレーション、オーム社、2018
- 藤原邦男、基礎物理学Ⅰ　物理学序論としての力学、東京大学出版会、1984
- 山本義隆、駿台受験シリーズ　新・物理入門　増補改訂版、駿台文庫、1987
- 今井　功ほか、セミナーライブラリー物理学=2　演習力学[新訂版]、サイエンス社、1981

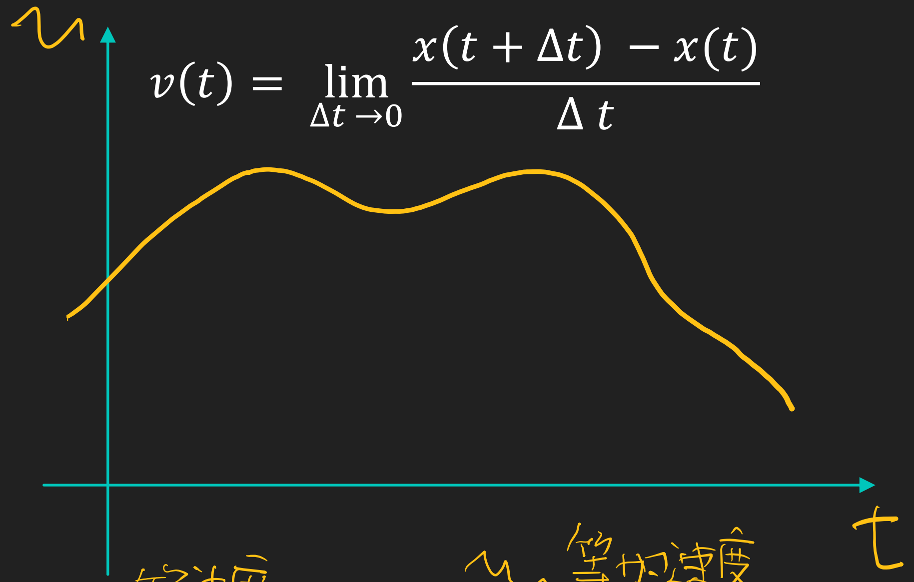
float 型 (小数の型)

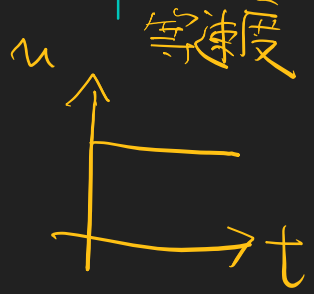
- int: 整数の型
  - 文字 => 整数への変換　int(　) 関数
- float: 小数の型
  - 文字 => 小数への変換　float(　) 関数
- String: 文字の型
  - 数字 => 文字への変換　str(　) 関数

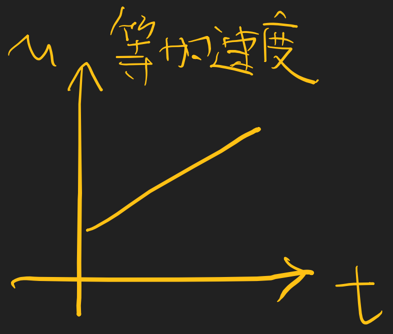
物理の理論とプログラミングの違い

- 物理（理論）

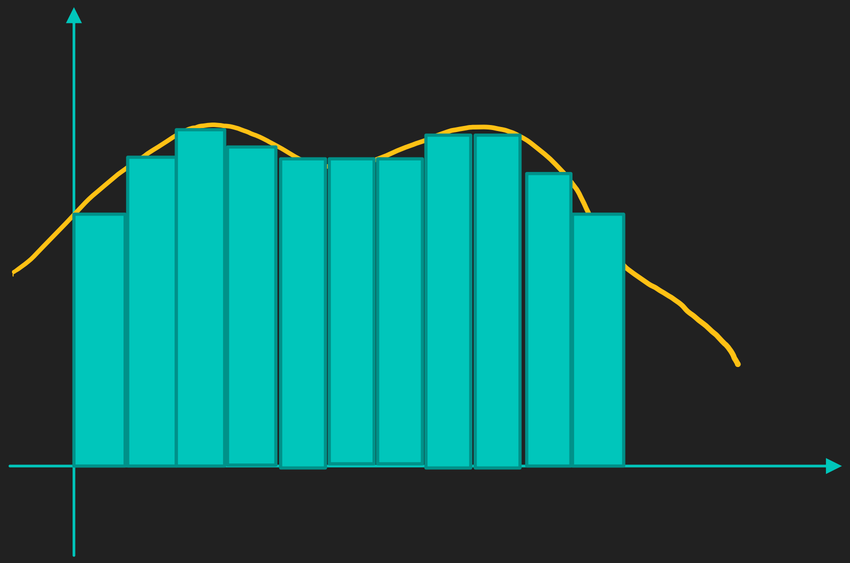
$$v(t) = \lim_{\Delta t \to 0} \frac{x(t+\Delta t) - x(t)}{\Delta t}$$







- プログラミング



Δtをある程度の小ささで妥協して，
運動の振る舞いを観測してみる

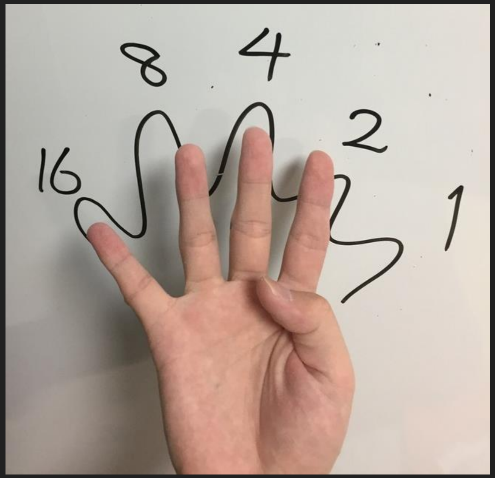
離散的なデータ

2進数

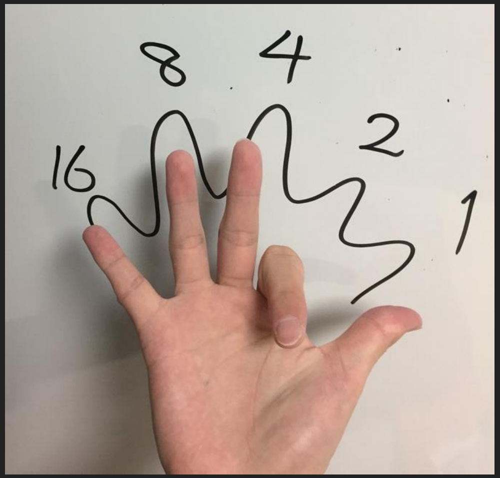
1	0	1	1
8の位	4の位	2の位	1の位

2進数の簡単数え方

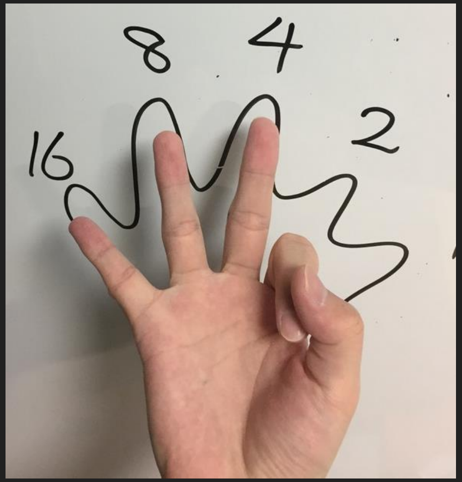
1



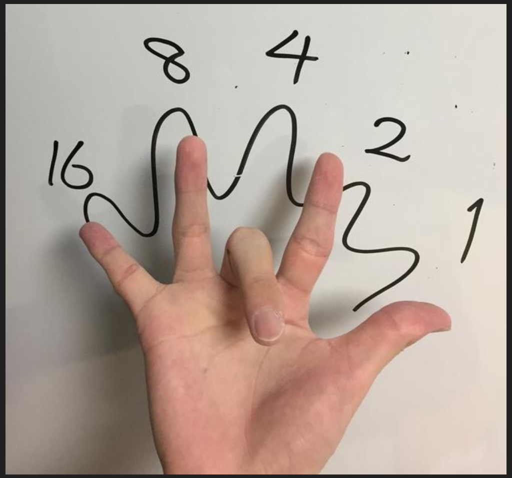
2



3

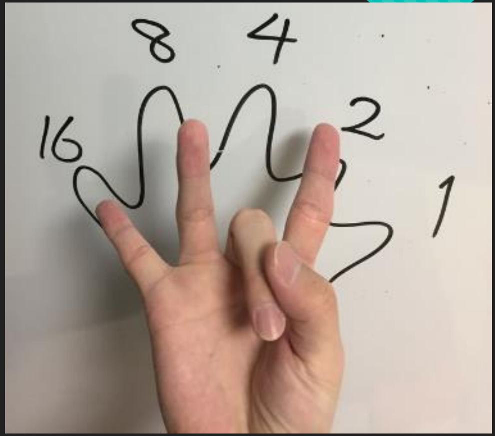


4

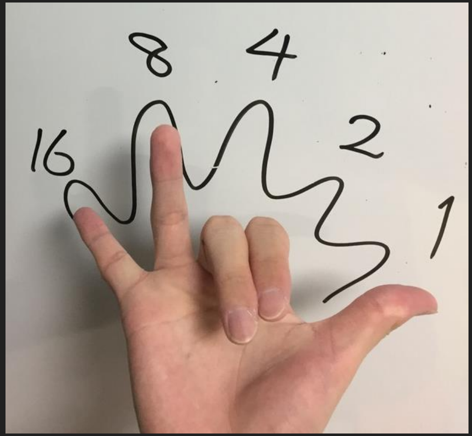


例続き

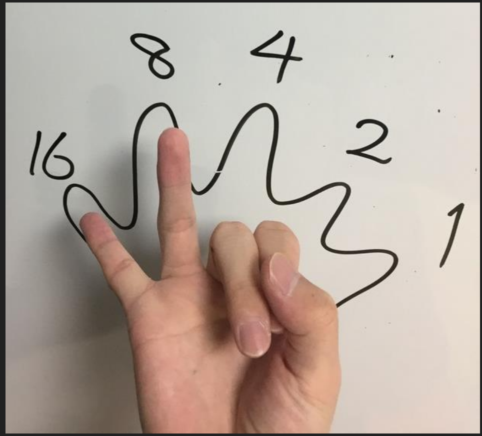
5



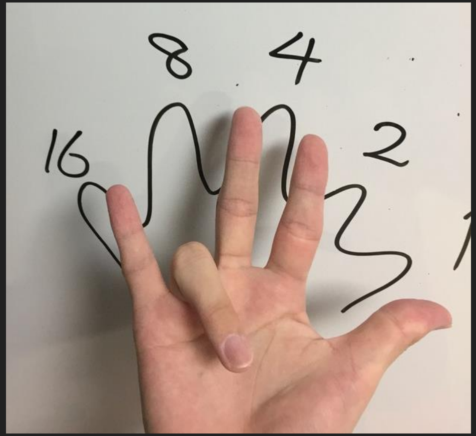
6



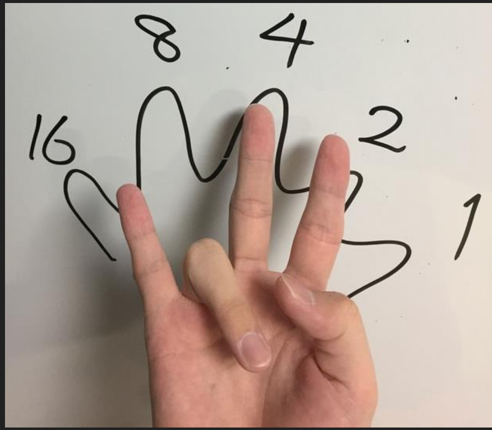
7



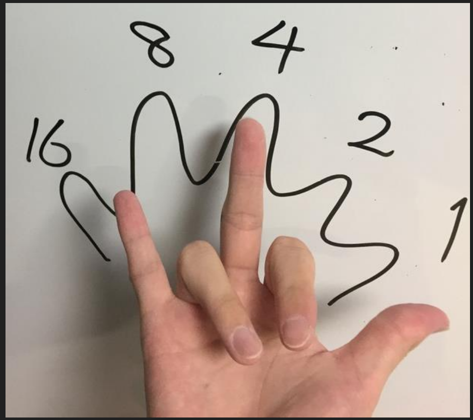
8



9



10



Q: この後の指の折り方はどうなるだろうか?

Q: いくつまで数えられそうか?

Q: 指の折り方に規則性はあるか?

2進数の小数

0	0	1	1	.	1	0	1	0
8の位	4の位	2の位	1の位		$1/2$の位	$1/4$の位	$1/8$の位	$1/16$の位

3 . 625

2進数の小数
コンピュータが普段していること

0	0	1	1	.	1	0	1	0
8の位	4の位	2の位	1の位		$^1/_2$の位	$^1/_4$の位	$^1/_8$の位	$^1/_{16}$の位

3.625

入力された10進数を2進数に変換してメモリに格納
=> $^1/_{2n}$ の和でうまく表せない小数はどうなる?

小数と無限小数

- Python での小数の表し方

$$\mathrm{a} \times 2^{n}$$

a: 1. XXXXXXXXXXXの形で表記した二進数
n: 指数部

aとnをペアで保存して小数を表す

ex)
$1.00 \times 2^2 = 1 \times 2^2 = 4$

$\Rightarrow$ ${}^{1}/_{2^n}$ の和でうまく表せない小数はどうなる?

$\Rightarrow$ 近似するしかない

バグの原因 誤差

- 0.06 秒経過後に何か処理をしたい場合

```python
t = 0.0

dt = 0.01

for i in range(50):
  t += dt
  print(t)

  if (t == 0.06):
    print("hit")
```

```
0.01
0.02
0.03
0.04
0.05
0.060000000000000005
0.07
0.08
0.09
0.09999999999999999
0.10999999999999999
0.11999999999999998
0.12999999999999998
0.13999999999999999
```

改善案（本日使います）

- 意図的に結果を四捨五入する（切り捨てもOK）

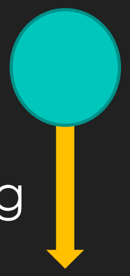
```python
t = 0.0

dt = 0.01

for i in range(50):
  t += dt
  print(t)

  if (round(t, 2) == 0.06):
    print(round(t, 2))
    print("hit")
```

```
0.01
0.02
0.03
0.04
0.05
0.060000000000000005
0.06
hit
0.07
0.08
0.09
```

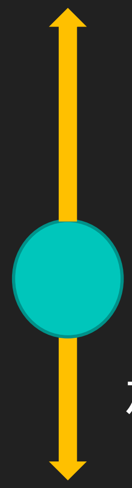
演習

- ロケットの逆噴射
  - 初期速度v0, 初期高度x0, 逆噴射の開始時刻t_fをユーザが指定して、ロケットの着陸シミュレーションを行う

逆噴射前 t < t_f



逆噴射開始後 t > t_f



演習のプログラム例

https://colab.research.google.com/drive/1aS643l6jy8FU0j4OqSYqtEbKlZHl46DT?usp=sharing

```
# -*- coding: utf-8 -*-
# 9/17 用演習プログラム
import numpy as np
import matplotlib.pyplot as plt

# 定数
g = 9.8

# tf 秒後に逆噴射の加速度を返す関数

def retrofire(t, tf):
    if t >= tf:
        return -1.5 * g  #重力の1.5倍の加速度で逆噴射を行う（ロケットの重さは考えない）
    else:
        return 0.0;

t = 0.0    # 時刻t
dt = 0.01  # 時刻の刻み幅

# 係数の入力
v = float(input("初速度v0を入力してください:"))
x0 = float(input("初期高度x0を入力してください:"))
tf = float(input("開始時刻tfを入力してください:"))
x = x0  # 初期高度の設定
```

時間経過後に、 逆噴射分の加速度を返す関数

続き（前回の復習+α）

```python
tlist = [t]
xlist = [x]
# 自由落下の計算
while (x > 0) and (x <= x0):
    t += dt                            # 時刻の更新
    # ここに記入                                 # 速度の計算
    # ここに記入                                 # 位置の更新

    tlist.append(t)
    xlist.append(x)
# グラフの表示
plt.plot(tlist, xlist)  # グラフをプロット
plt.show()
```

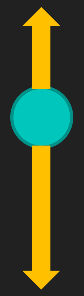
できたかなと思ったら

- v0 = 0, x0 = 100, tf = 2.62 として、滑らかな高度のグラフが描かれるかどうか試してください
  - 自由落下と同様の軌道を描いた場合　=> 着陸失敗
  - 初期高度 x0 を超えてしまう場合 => 着陸できずに再び空へ飛び立ちます
- 上記の条件でうまくいったら、tf や v0などの値をいろいろ試して、どのような場合に着陸失敗するのか、空へ飛び立つのかを調べてみてください。

雨粒の運動

- 雨はどのようにして降ってくるのか？
  - 空で雨粒ができてから、重力で加速し続ける？
  - 何らかの力が働いて、遅くなる？
  - 何らかの力が働いて、一定になる？

雨粒にかかる力



Q: 加速度はどうなる？

運動方程式

$$
\begin{aligned}
ma &= F \\
ma &= mg - kv \\
a &= g - \frac{k}{m}v
\end{aligned}
$$

演習　シミュレーション

雨粒の重さm = 0.0000005 [kg]

空気抵抗の定数k = 1.089× 10^{-5} $[kg・m^{-1}・s^{-1}]$

重力加速度 g = 9.8 $[m・s^{-2}]$

初期高度 x0 = 2000 [m]

として、時間における雨水の速度の変化をグラフ化してください。

前回のプログラムを利用してください。

加速度の計算には、1ループ前の速度を利用可能であると仮定します。

ある程度たったら答えのグラフを見せます。

補足: 加速度の定義と，微分積分の関係

$$a(t) = \lim_{\Delta t \to 0} \frac{v(t+\Delta t) - v(t)}{\Delta t} = v'(t)$$

- 加速度も，速度と同様の議論で上の式を導ける(p. 3-9). （興味があれば考えてみてください）

tで微分

$$\begin{aligned} x(t) \quad &= \frac{1}{2} g t^2 + v_0 t + x_0 \\ v(t) = x'(t) &= g t + v_0 \\ a(t) = v'(t) &= g \end{aligned}$$

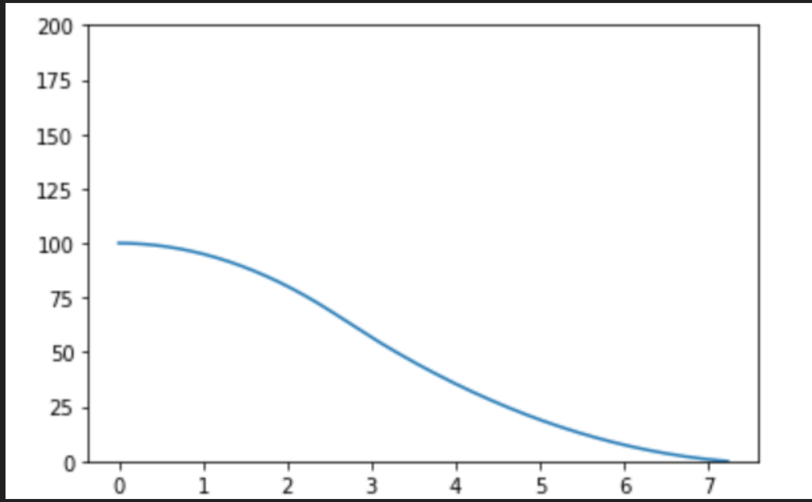
tで積分

Q(最終課題用): 加速度から積分して導いた後，積分定数をどのように計算すれば
上の公式が導けるだろうか?(応用)

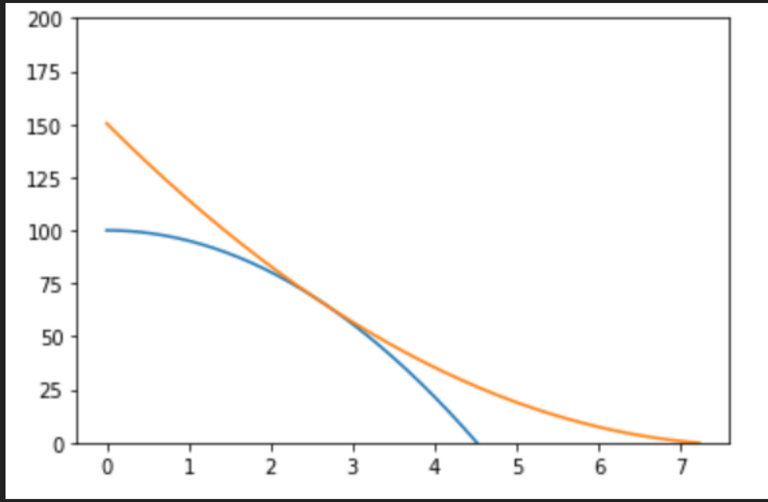
演習（応用）

- ロケットの逆噴射で作成したコードに、1つ前のページにならって高度xを表す式を付け足し、シミュレーションで作られた軌道を再現するグラフを記述してください。（下の右側グラフが応用、やや難しいです。）

シミュレーションで作ったグラフ



公式に当てはめて作ったグラフ



自由課題（簡単なもの）

1. 等加速度運動の公式を描画するプログラムを作成してください。物理で習った式をそのまま使って出力すればOKです

2. 1のプログラムと、前回作ったシミュレーション上での等加速度運動の公式のグラフを比較して、一致することを確認してください。（初期条件はすべて揃えてください）

3. 万有引力$F = \frac{G\, m_1 m_2}{r^2}$のrの変化による推移を確かめるグラフ($r - F$グラフ)を作成してください
   1. G = 6.674×10^{-11}, $m1 = 1.5, m2 = 0.5$ として確認してみてください。
   2. できたらm1, m2の値を変えて大きさを確認してみてください。
      （地球とリンゴの重さ、地球と月の重さを調べてみるなど）