電車の待ち時間を平等にする

チーム名：でんでんむし

きっかけ

1限に間に合うように通学すると、通勤ラッシュと被ることが辛い。
特に池袋駅は利用者が多く、毎日混雑している。

↓疑問

「駅ごとの待ち時間に差はないの？」
「今の運転間隔や人数制限のない状態は効率がいいの？」

研究内容

今回の研究では、乗客の待ち時間の公平さに注目し

各駅の平均待ち時間が等しくなるような

「運転間隔」を調べることにした。

設定

シミュレーションでは以下の状況を考えることにした。

・運転間隔時間 k ただし、kは $3 \leq k \leq 30$の変数
・駅の数 $j = 20$(駅)
・電車の定数 $M = 300$(人)
・各駅に来る最大の人数 $v =$ 運転間隔時間 $\times 10$ (人)
$= k \times 10$
・観察時間 $T - length = 120$(分)
・各駅で電車から降りる人の割合
$r = 50$ (%)
・始発時に i 駅 ($1 \leq i \leq 20$) のホームにいる乗客の数
$Wait(i = 1) = 70$ (始発駅)
$Wait(i \neq 1) = 50$

設定

次の電車が来るまでにホームに来る人数を、

$k \times 10 - 10$(人)〜$k \times 10 + 10$(人)

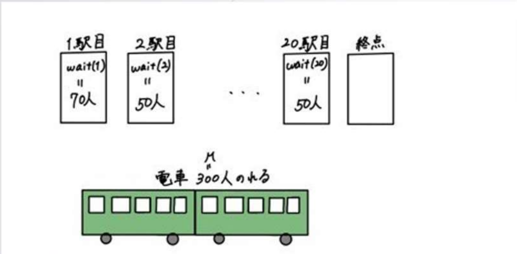
の範囲でランダムに定める。

電車の本数は、

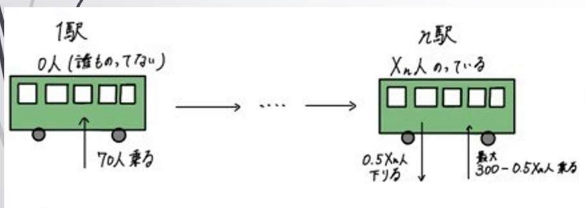
電車の本数($kaisu$) $= T - length \,/\, k$(小数点切り捨て)

とする。

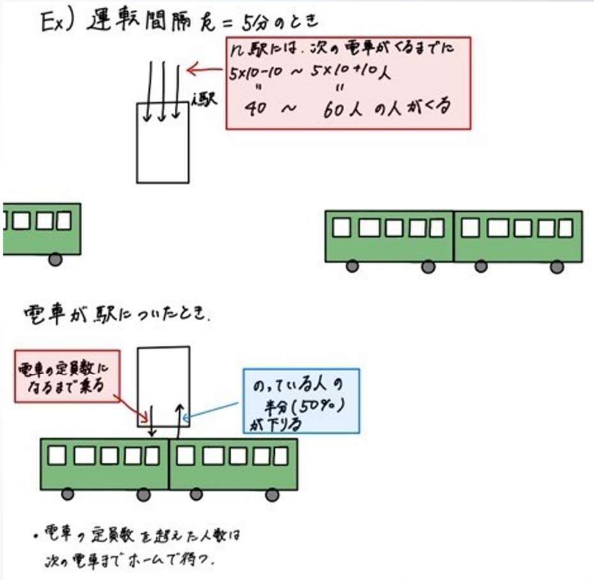
設定



電車内の人の動き



ホームにいる人の動き



実験1

駅ごとの利用者と待ち時間の合計をそれぞれ求め、
駅ごとの平均待ち時間を求めるプログラミングを作る。

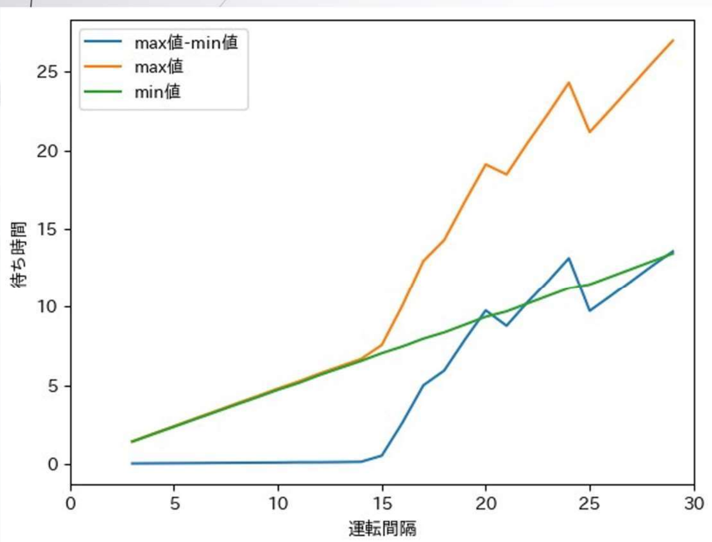
このシミュレーションを100回行い平均を取る。

その平均待ち時間が均等になるときの運転間隔k (分)を求める。

実験1のコード

```
from re import X
import random
M=300 #電車の乗員数
r=0.5 #降りる人の割合
T_length=120 #観察時間
zero=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
print('運転間隔','駅ごとの平均の差','駅の待ち時間の平均最
大値','駅の待ち時間の平均最小値')
for k in range(3,30):
  kaisu=int(T_length/k) #電車の本数
  v=k*10 #駅に来る人の最大値
  x=0 #電車に乗っている人
    tsum=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
#駅ごとの合計待ち時間
  hsum=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #
駅ごとの利用者数
  for sim in range(0,100):
    wait=[70,50,50,50,50,50,50,50,50,50,50,50,50,5
0,50,50,50,50,50,50]
     #[30,20,20,20,20,20,20,20,20,20,20,20,20,20,2
0,20,20,20,20,20]
    for j in range(0,kaisu):
      if j==0:#始発
        w=0
        x=0
        for i in range(0,20):
          x=int((1-r)*x) #2割降りたあとの人数
          hsum[i]+=wait[i]
          if wait[i]+x<=M:#待ち人と乗ってる人の合計が乗員数より
少ないとき
            x+=wait[i]#乗った人
            wait[i]=0
          else:#待ち人と乗ってる人の合計が乗員数より多いとき
            wait[i]=wait[i]-M+x
            x=M
      else:#始発以外の時
        x=0#最初は電車に0人
        for i in range(0,20):
          x=int((1-r)*x)
          w=random.randint(v-10,v+10)#最大50人乗ってくる
          tsum[i]+=int(k*(wait[i])+(w*k)/2)#j-1番目の電車か
らj番目の電車までの待ち時間　変えた
          hsum[i]+=w
          if wait[i]+x+w<=M:
            x=wait[i]+x+w
            wait[i]=0
          else:
            wait[i]+=w+x-M
            x=M
  result = [x / y for x, y in zip(tsum, hsum)]
  print(k,max(result)-
min(result),max(result),min(result))
```

結果 (実験1)



運転間隔	駅ごとの平均の差	駅の待ち時間の平均最大値	駅の待ち時間の平均最小値
3	0.0234328803023704	1.43127911728647	1.40784623698410
4	0.0322472993541076	1.91786447638603	1.88561717703192
5	0.0398738212248548	2.39189401910506	2.35202019788020
6	0.0476594387147213	2.87430343151631	2.82664399280159
7	0.0568565524663995	3.34762071229419	3.29076415982779
8	0.0653233157801209	3.82999557988507	3.76467226410495
9	0.0764636898826944	4.29940056510602	4.22293687522332
10	0.0814755548429300	4.78327582918667	4.70180027434374
11	0.0987476049751663	5.23418006770039	5.13543246272522
12	0.0994478936624840	5.73532603421352	5.63587814055103
13	0.1123298154745740	6.20075428173887	6.08842446626430
14	0.1288598510112280	6.66112853878625	6.53226868777503
15	0.5200146599043860	7.54984968347244	7.02983502356806
16	2.6350545290253500	10.09220320243240	7.45714867340707
17	4.9879093726999300	12.94055558669130	7.95264621399139
18	5.9254884452712700	14.27633183366270	8.35084338839143
19	7.8945638705702500	16.74147333579250	8.84690946522229
20	9.7538559380179600	19.10084649736710	9.34699055934923
21	8.7707463405611000	18.46019567976810	9.68944933920704
22	10.2598569464486000	20.45103341703680	10.19117647058820
23	11.6477110512696000	22.33341581096120	10.68570475969160
24	13.1206445200334000	24.30604377137660	11.18539925134310
25	9.7248359551780500	21.15755627009640	11.43272031491840
26	10.6702164602765000	22.59825359024320	11.92803712996660
27	11.6444396721092000	24.07003927218640	12.42559960007720
28	12.6270059235325000	25.55000000000000	12.92299407646740
29	13.5606437805143000	26.98095227746870	13.42030849695430

わかったこと

実験1の結果から、運転間隔が3分から16分のとき平均待ち時間の差が1分以下になった。

また、16分以降では平均待ち時間の差は一次関数にはならず、かなりギザギザしたグラフになった。

最小の平均待ち時間は一次関数になるが、最大の平均待ち時間は17分から急激に増加している。

21分、25分では最大の平均待ち時間が下がっていることがわかり、何か見落としている部分があると考える。

問題点

この実験の問題点として、
最後の電車に乗れなかった人が存在することが挙げられる。

そこで

ホームに残っている人がいなくなるまで電車を追加し
平均時間を見ることにした。
ただし、$T - length \,/\, k$ 本目の電車が通過した後、
駅のホームに新しく来る人はいないものとする。

実験2

実験1のコードにホームで待っている人が0になるまで
電車が来るように変更した。

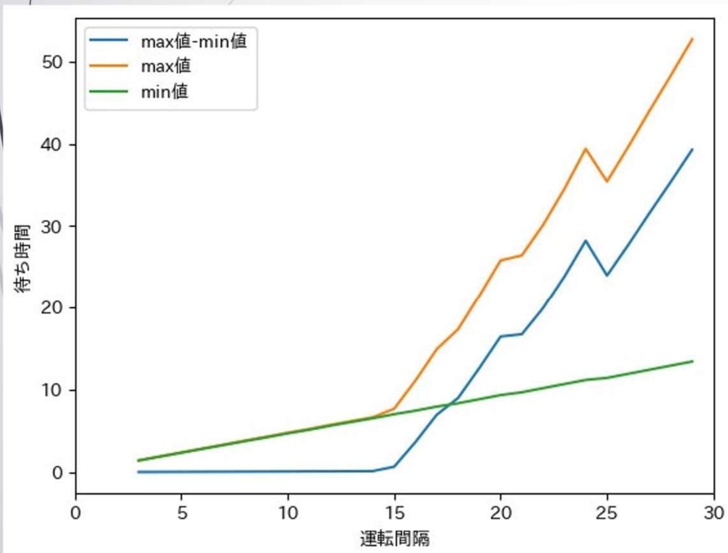
このシミュレーションを100回行い平均を取る。

平均待ち時間が均等になるときの運転間隔k (分)を求める。

実験2のコード

```
from re import X
import random
M=300 #電車の乗員数
r=0.5 #降りる人の割合
T_length=120 #観察時間
zero=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
print('運転間隔','駅ごとの平均の差','駅の待ち時間の平均最大値','駅の待ち
時間の平均最小値')
for k in range(3,30):
  kaisu=int(T_length/k) #電車の本数
  v=k*10 #駅に来る人の最大値
  x=0 #電車に乗っている人
    tsum=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #駅ごとの合計
待ち時間
  hsum=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #駅ごとの利用者
数
  for sim in range(0,100):
    wait=[70,50,50,50,50,50,50,50,50,50,50,50,50,50,50,50,50,5
0,50,50]
    for j in range(0,kaisu):
      if j==0:#始発
        w=0
        x=0
        for i in range(0,20):
          x=int((1-r)*x) #2割降りたあとの人数
          hsum[i]+=wait[i]
          if wait[i]+x<=M:#待ち人と乗ってる人の合計が乗員数より少ないとき
            x+=wait[i]#乗った人
            wait[i]=0
          else:#待ち人と乗ってる人の合計が乗員数より多いとき
            wait[i]=wait[i]-M+x
            x=M
      else:#始発以外の時
        x=0#最初は電車に0人
        for i in range(0,20):
          x=int((1-r)*x)
          w=random.randint(v-10,v+10)#最大50人乗ってくる
          tsum[i]+=int(k*(wait[i])+(w*k)/2)#j-1番目の電車からj番目の
電車までの待ち時間　変えた
          hsum[i]+=w
          if wait[i]+x+w<=M:
            x=wait[i]+x+w
            wait[i]=0
          else:
            wait[i]+=w+x-M
            x=M
    while wait!=zero:
      x=0
      for i in range(0,20):
        x=int((1-r)*x)
        if wait[i]!=0:
          tsum[i]+=int(k*(wait[i]))
          if wait[i]+x<=M:
            x=wait[i]+x
            wait[i]=0
          else:
            wait[i]+=x-M
            x=M
  result = [x / y for x, y in zip(tsum, hsum)]
  print(k,max(result)-min(result),max(result),min(result))
```

結果 (実験2)



運転間隔	駅ごとの平均の差	駅の待ち時間の平均最大値	駅の待ち時間の平均最小値
3	0.0229513782594250	1.43118756029366	1.40823618203423
4	0.0312341111735090	1.91761342571614	1.88637931454263
5	0.0397495833915519	2.39182700295749	2.35207741956594
6	0.0477255485195868	2.87439815783964	2.82667260932006
7	0.0570640601932868	3.34777239317622	3.29070833298293
8	0.0664497170403772	3.82996522788910	3.76351551084872
9	0.0763475078566049	4.29928749460556	4.22293998674896
10	0.0821894093296657	4.78340163401807	4.70121222468840
11	0.0996415495739331	5.23401274374328	5.13437119416935
12	0.1004408760909790	5.73529333909805	5.63485246300707
13	0.1123511120457120	6.20106637888460	6.08871526683888
14	0.1282697721145700	6.66133183674259	6.53306206462802
15	0.6556146680642880	7.68506060193381	7.02944593386952
16	3.6504915162135100	11.10725050151690	7.45675898530339
17	6.9622319131938800	14.91467318766850	7.95244127447470
18	8.9862816681191100	17.33754618743700	8.35126451931790
19	12.6229540316602000	21.47115932040870	8.84820528874844
20	16.4571721343242000	25.80251377246640	9.34534163814226
21	16.7384758779382000	26.42836183628440	9.68988595834617
22	19.9425248202976000	30.13216057735670	10.18963575705910
23	23.8326365550707000	34.51897678727670	10.68634023220590
24	28.2098827035864000	39.39411904197290	11.18423633838650
25	23.9734330594030000	35.40760178513550	11.43416872573240
26	27.7030310862882000	39.63361292028160	11.93058183399330
27	31.5970899576387000	44.02339391549420	12.42630395785550
28	35.3860720585159000	48.30952754669590	12.92345548817990
29	39.2871361271515000	52.70735064597240	13.42021451882080

わかったこと

0分～15分では平均待ち時間の差が1分以下となり、実験1と大差ないことがわかり、最小の平均待ち時間は実験1より緩やかに上昇することがわかった。

また、16分以降の平均待ち時間の差は実験1と比べて一次関数に近くなったが、運転間隔が21分と25分のとき下がってしまうという点は解消されなかった。

問題点

この実験の問題点として、
観察時間120分に対し、最終電車が通過した後に人が来ないことが挙げられる。
例えば、運転間隔が25分のとき電車の本数は$120 \div 25 \fallingdotseq 4$(本)になる。
この場合、最後の電車が来るまでにかかる時間は$4 \times 25 = 100$(分)となり、 20分間人が来ない時間がある。

そこで

最終電車が来た後も計測時間(120分)の間はホームに人が来るようにする。

実験3

実験2のコードに計測時間が終わるまでは**ホームに人が来るよう設定(☆)**し、このシミュレーションを100回行い平均を取る。

平均待ち時間が均等になるときの運転間隔k (分)を求める。

(☆) のホームに来る人数の決め方
最後電車の通過後から計測時間が終わるまでにホームに来る人数を

$$(120 - kaisu \times k) \times 10 - 10(\text{人}) \sim (120 - kaisu \times k) \times 10 + 10(\text{人})$$

の範囲でランダムに定める。

※電車の本数$(kaisu) = T - length\ /\ k$(小数点切り捨て)

実験3のコード

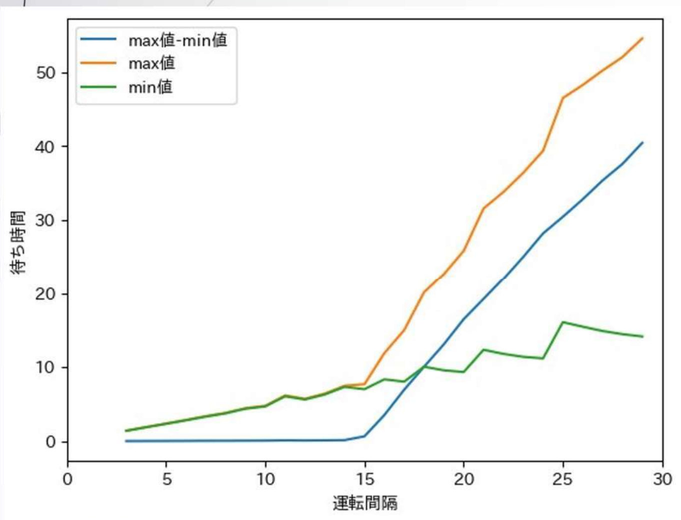
```
from re import X
import random
M=300 #電車の乗員数
r=0.5 #降りる人の割合
T_length=120 #観察時間
zero=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
print('運転間隔','駅ごとの平均の差','駅の待ち時間の平均最大値','駅の待ち
時間の平均最小値')
for k in range(3,30):
  kaisu=int(T_length/k) #電車の本数
  v=k*10 #駅に来る人の最大値
  x=0 #電車に乗っている人
    tsum=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #駅ごとの合計
待ち時間
  hsum=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #駅ごとの利用者
数
  for sim in range(0,100):
    wait=[70,50,50,50,50,50,50,50,50,50,50,50,50,50,50,50,50,5
0,50,50]
    for j in range(0,kaisu):
      if j==0:#始発
        w=0
        x=0
        for i in range(0,20):
          x=int((1-r)*x) #2割降りたあとの人数
          hsum[i]+=wait[i]
```

```
          if wait[i]+x<=M:#待ち人と乗ってる人の合計が乗員数より少ないとき
            x+=wait[i]#乗った人
            wait[i]=0
          else:#待ち人と乗ってる人の合計が乗員数より多いとき
            wait[i]=wait[i]-M+x
            x=M
      else:#始発以外の時
        x=0#最初は電車に0人
        for i in range(0,20):
          x=int((1-r)*x)
          w=random.randint(v-10,v+10)#最大50人乗ってくる
          tsum[i]+=int(k*(wait[i])+(w*k)/2)#j-1番目の電車からj番目の
電車までの待ち時間　変えた
          hsum[i]+=w
          if wait[i]+x+w<=M:
            x=wait[i]+x+w
            wait[i]=0
          else:
            wait[i]+=w+x-M
            x=M
```

実験3のコード

```
    for b in range(0,20):
      m=(120-kaisu*k)*10
      if m-10<0 and m!=0:
        w=random.randint(0,m+10)#最大50人乗ってくる
        wait[b]+=w
        hsum[b]+=w
        tsum[b]+=(w*(120-kaisu*k))/2
      elif m-10>=0:
        w=random.randint(m-10,m+10)#最大50人乗ってくる
        wait[b]+=w
        hsum[b]+=w
        tsum[b]+=(w*(120-kaisu*k))/2
      else:
        pass
    while wait!=zero:
      x=0
      for i in range(0,20):
        x=int((1-r)*x)
        if wait[i]!=0:
          tsum[i]+=int(k*(wait[i]))
          if wait[i]+x<=M:
            x=wait[i]+x
            wait[i]=0
          else:
            wait[i]+=x-M
            x=M
  result = [x / y for x, y in zip(tsum, hsum)]
  print(k,max(result)-min(result),max(result),min(result))
```

結果



運転間隔	駅ごとの平均の差	駅の待ち時間の平均最大値	駅の待ち時間の平均最小値
3	0.0239970660698818	1.43144695057405	1.40744988450417
4	0.0315028794861991	1.91778344158513	1.88628056209893
5	0.0388673938336396	2.39158069883527	2.35271330500163
6	0.0484449719163535	2.87478504766515	2.82634007574880
7	0.0627698960603733	3.38547100467448	3.32270110861410
8	0.0646205407838955	3.82963644417186	3.76501590338796
9	0.0766444934669845	4.46475481046302	4.38811031699604
10	0.0826714069548373	4.78349542309324	4.70082401613840
11	0.1203102779152640	6.18665754601980	6.06634726810453
12	0.1001448782130990	5.73530501685224	5.63516013863914
13	0.1212852615791660	6.43249174285790	6.31120648127873
14	0.1510526490551660	7.49325028760425	7.34219763854909
15	0.6662752384549160	7.69509205795838	7.02881681950346
16	3.5334122559489400	11.88932965285260	8.35591739690373
17	6.9576654361173200	15.00144977052990	8.04378433441263
18	10.0517858671686000	20.11528389976600	10.06349803259740
19	13.0919613205475000	22.66894111225090	9.57697979170337
20	16.4868968923589000	25.83228750713060	9.34539061477172
21	19.2267019020661000	31.57408969762040	12.34738779555420
22	22.0042298017401000	33.79907835885910	11.79484855711900
23	25.0068964895713000	36.40071847507330	11.39382198550190
24	28.2031234896878000	39.38800102724160	11.18487753755380
25	30.4543874390075000	46.52939062375210	16.07500318474450
26	32.8275552366151000	48.28934163593490	15.46178639931970
27	35.3712998152825000	50.25266712093320	14.88136730565070
28	37.6061191258632000	52.06161454845190	14.45549542258860
29	40.4570953804713000	54.59630102703090	14.13920564655960

わかったこと、考察

運転間隔が25分のときに待ち時間の差が急激に下がるということがなくなり、26分以降の最大の平均待ち時間は一次関数になった。

一方で、実験1、2のときとは違い最小の平均待ち時間の方は減ったり増えたりと一次関数から少し離れたグラフになることがわかった。
減る時間は0.3分ほどのため、人数をランダムにしたことが原因ではないかと考えられるが、増える時間は5分と大きいため見落としている条件がまだあるのではと考えられる。

まとめ

実験1、2、3より、平均待ち時間を均等にするには、

3分～15分

が良いと考えられる！

それ以降では、終点に近い駅に人が溜まり差が大きくなってしまうため、待ち時間は平等にはならなくなってしまう。

アニメーションの作成

運転間隔の差でホームの人数はどう変わるかを可視化するため、下記の設定でシミュレーションを行いアニメーションを作成した。
コードは実験2のものを採用した。

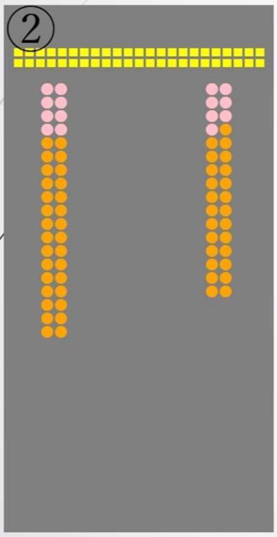
- 運転間隔 $k = 5, 8$
- 駅の数 $j = 6$(駅)
- 電車の乗員数 $M = 70$(人)
- 各駅に来る最大の人数 $v = k \times 10$ (人)
- 観察時間 $T - length = 40$(分)
- 降りる人の割合 $r = 50$ (%)
- 始発時に i 駅 $(1 \le i \le 5)$ のホームにいる乗客の数

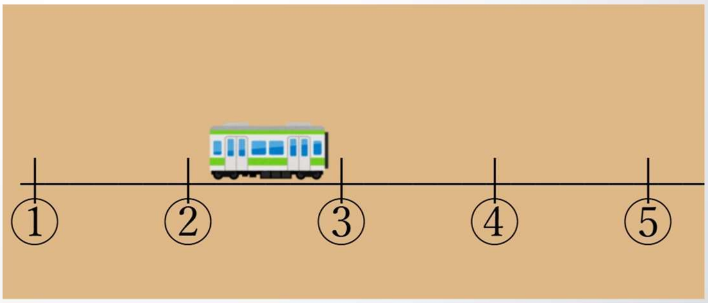
$Wait(i = 1) = 70$ (始発駅)

$Wait(i \neq 1) = 50$

アニメーションの作成

ホームで待つ人がどの電車の後に来たのかわかるように色を変えた。
●：始発前に来た人
●：1本目と2本目の電車の間に来た人
ⓘ: 駅の番号(i = 1,2,3,4,5)





感想

最初の予想では、運転間隔が短いほど待ち時間は均等になると予想していたが、15分までは差がほとんどないことに驚いた。

今回は運転間隔に絞って考えたが、待っている人の人数で乗車人数を決めるとどうなるのかや、初期の設定を変えた場合どうなるのか気になった。

また、この研究を極めれば”利用者から運転間隔を決める”ということもできるのではないかと考えた。