

令和4年度総合型選抜プログラミング実技試験
事前公開プログラム

日本大学文理学部情報科学科

このプログラムは、以下のモデルに従って感染症の感染者数をシミュレーションするものである。

- 人々の状態は感染可能、感染中、免疫獲得のいずれかであり、各状態の総人口を1とした割合をそれぞれ s, i, r とする。
- 一定の期間につき、感染可能者のうちの一定の割合 β が感染者になる。
- 一定の期間につき、感染者のうちの一定の割合 μ が免疫獲得者になる。
- 総人口は変化せず、上記以外の状態の変化は起こらない。

Java Mode

```
1 void setup() {
2   size(600, 600);
3 }
4
5 void draw() {
6   background(255);
7   float beta = 0.01;
8   float mu = 0.005;
9   float t0 = 0;
10  float s0 = 1;
11  float i0 = 0;
12  float r0 = 0;
13  while (t0 < width) {
14    float t = t0 + 1;
15    float s = s0 - beta * s0;
16    float i = i0 - mu * i0 + beta * s0;
17    float r = r0 + mu * i0;
18
19    stroke(255, 0, 0);
20    line(t0, height * (1 - s0), t, height * (1 - s));
21    stroke(0, 255, 0);
22    line(t0, height * (1 - i0), t, height * (1 - i));
23    stroke(0, 0, 255);
24    line(t0, height * (1 - r0), t, height * (1 - r));
25
26    t0 = t;
27    s0 = s;
28    i0 = i;
29    r0 = r;
30  }
31 }
```

Python Mode

```
1 def setup():
2     size(600, 600)
3
4 def draw():
5     background(255)
6     beta = 0.01
7     mu = 0.005
8     t0 = 0.0
9     s0 = 1.0
10    i0 = 0.0
11    r0 = 0.0
12    while t0 < width:
13        t = t0 + 1
14        s = s0 - beta * s0
15        i = i0 - mu * i0 + beta * s0
16        r = r0 + mu * i0
17
18        stroke(255, 0, 0)
19        line(t0, height * (1 - s0), t, height * (1 - s))
20        stroke(0, 255, 0)
```

```
21     line(t0, height * (1 - i0), t, height * (1 - i))
22     stroke(0, 0, 255)
23     line(t0, height * (1 - r0), t, height * (1 - r))
24
25     t0 = t
26     s0 = s
27     i0 = i
28     r0 = r
```

p5.js Mode

```
1  function setup() {
2    createCanvas(600, 600);
3  }
4
5
6  function draw() {
7    background(255);
8    const beta = 0.01;
9    const mu = 0.005;
10   let t0 = 0;
11   let s0 = 1;
12   let i0 = 0;
13   let r0 = 0;
14   while (t0 < width) {
15     const t = t0 + 1;
16     const s = s0 - beta * s0;
17     const i = i0 - mu * i0 + beta * s0;
18     const r = r0 + mu * i0;
19
20     stroke(255, 0, 0);
21     line(t0, height * (1 - s0), t, height * (1 - s));
22     stroke(0, 255, 0);
23     line(t0, height * (1 - i0), t, height * (1 - i));
24     stroke(0, 0, 255);
25     line(t0, height * (1 - r0), t, height * (1 - r));
26
27     t0 = t;
28     s0 = s;
29     i0 = i;
30     r0 = r;
31   }
32 }
```