

2017 年度情報メディア基盤ユニット Advanced Topics 複数キーの同時押しを拾う

プログラミングの経験がある人に向けた資料です。質問があれば佐藤まで。

Processing でプログラムを作っていると、複数キーの同時押しなどに対応したいことがあります。keyPressed などでは対応することが困難なので、KeyboardObserver というクラスを作って対応することを行っていました。キーが押されているかを記憶している boolean 型の配列を用意しておき、Processing の keyPressed や keyReleased を利用して、押されたキーの状況を管理するようにしています。

8~43 行が KeyboardObserver クラスの定義です。利用するためには、45 行目のように KeyboardObserver のインスタンスを取得し、boolean doesKeyPress(char)関数と boolean doesSpecialKeyPress(int)関数を利用することで、指定されたキーが押されているかを確認することが出来ます。後者はカーソルキーなどを特殊キーが押されているかを判定するものです。キーの押下の情報は、Processing の keyPressed 関数 (50~52 行目) と keyReleased 関数 (54~56 行目) の中で行っています。細かいことはプログラムを見て下さい。

複数のキー押下に対応したサンプルプログラム

```
1  /*
2  複数キーの押下をチェックするサンプルプログラム
3  カーソルキーを移動すると円が移動、
4  r, g, b のキーを押すと円の色が変換
5
6  Programmeed by H. Sato
7  */
8  static class KeyboardObserver {
9      private static KeyboardObserver theKeyboardObserver = new KeyboardObserver ();
10     boolean[] normalKeys;
11     boolean[] specialKeys;
12     private KeyboardObserver () {
13         normalKeys = new boolean[256];
14         specialKeys = new boolean[256];
15         for(int i=0;i<256;i++) {
16             normalKeys[i] = false;
17             specialKeys[i] = false;
18         }
19     }
20     public static KeyboardObserver getKeyboardObserver () {
21         return theKeyboardObserver;
22     }
23     void pressKey(char key, int keyCode) {
24         if(key == CODED) {
25             specialKeys[keyCode] = true;
26         }else{
27             normalKeys[key] = true;
```

```
28     }
29     }
30     void releaseKey(char key, int keyCode) {
31         if(key == CODED) {
32             specialKeys[keyCode] = false;
33         }else{
34             normalKeys[key] = false;
35         }
36     }
37     boolean doesKeyPress(char k) {
38         return normalKeys[k];
39     }
40     boolean doesSpecialKeyPress(int k) {
41         return specialKeys[k];
42     }
43 }
44
45 KeyboardObserver keyboard = KeyboardObserver.getKeyboardObserver();
46 int x;
47 int y;
48 color fillColor=color(0,0,0);
49
50 void keyPressed() {
51     keyboard.pressKey(key, keyCode);
52 }
53
54 void keyReleased() {
55     keyboard.releaseKey(key, keyCode);
56 }
57
58 void setup() {
59     size(400,400);
60     smooth();
61     x = width/2;
62     y = height/2;
63 }
64
65 void draw() {
66     background(255);
67     stroke(0);
68     fill(fillColor);
69     update();
70     ellipse(x, y, 20, 20);
71 }
72
73 void update() {
74     if(keyboard.doesSpecialKeyPress(UP)) {
75         y--;
76     }
77     if(keyboard.doesSpecialKeyPress(DOWN)) {
78         y++;
79     }
80     if(keyboard.doesSpecialKeyPress(LEFT)) {
81         x--;
82     }
83     if(keyboard.doesSpecialKeyPress(RIGHT)) {
84         x++;
```

```
85     }  
86     if(keyboard.doesKeyPress(' i')) {  
87         x = width/2;  
88         y = height/2;  
89     }  
90     fillColor = color(0,0,0);  
91     if(keyboard.doesKeyPress(' r')) {  
92         fillColor |= color(255,0,0);  
93     }  
94     if(keyboard.doesKeyPress(' g')) {  
95         fillColor |= color(0,255,0);  
96     }  
97     if(keyboard.doesKeyPress(' b')) {  
98         fillColor |= color(0,0,255);  
99     }  
100 }
```