

キュークラス

古城隆人

2024 年 10 月 29 日

Listing 1: main class

```

1  public class newClass {
2
3  public static void main(String[] args) {
4      // Queue.defaultSize = 10;
5      Queue queue = new Queue();
6      System.out.println(queue.QueueVolume() + "個目のスタックです");
7      Queue tower10 = new Queue(10);
8      System.out.println(queue.QueueVolume() + "個目のスタックです");
9      tower10 = new Queue();
10     // System.out.println(queue.queueCount + "個目のスタックです");
11     queue.enqueue(10);queue.printQueue();
12     queue.enqueue(20);queue.printQueue();
13     // queue.volemu = 0;
14     queue.enqueue(30);queue.printQueue();
15     queue.enqueue(40);queue.printQueue();
16     queue.enqueue(50);queue.printQueue();
17     queue.enqueue(60);queue.printQueue();
18     // queue.data[0] = 100;
19     // queue.printQueue(0);
20     System.out.println(queue.dequeue());queue.printQueue(3,4);queue.printQueue();
21     System.out.println(queue.dequeue());queue.printQueue();
22     System.out.println(queue.dequeue());queue.printQueue();
23     System.out.println(queue.dequeue());queue.printQueue();
24     System.out.println(queue.dequeue());queue.printQueue();
25     System.out.println(queue.dequeue());queue.printQueue();
26 }
27
28 }
```

Listing 2: Queue class

```

1
2  public class Queue {
3      private int volume;
4      private int data[];
5      private final static int defaultSize = 5;
6      private static int queueCount = 0;
7
8      Queue() {
9          this(defaultSize);
10     }
11
12     Queue(int n) {
13         data = new int[n];
```

```

14         System.out.println(data.length + "個分のキュー生成");
15         queueCount++;
16         // System.out.println(queueCount + "個目のスタックです");
17     }
18
19     int enqueue(int number) {
20         int value;
21         // 残容量確認
22         if (data.length > volume) {
23             // 入力値確認
24             if (number > 0) {
25                 data[volume] = number;
26                 volume++;
27                 value = 1;
28             } else {
29                 value = 0;
30                 System.out.println("wrong input");
31             }
32         } else {
33             System.out.println("queue overflow");
34             value = 0;
35         }
36
37         return value;
38     }
39
40     // データ取得関数
41     int dequeue() {
42         int value;
43         // 格納個数確認
44         if (volume > 0) {
45             value = data[0];
46             volume--;
47
48             // 空き領域を埋めるためのシフト
49             for (int i = 0; i < data.length - 1; i++) {
50                 data[i] = data[i + 1];
51             }
52             data[volume] = 0;
53         } else {
54             value = -1;
55         }
56         return value;
57     }
58

```

```

59     // 状態表示関数
60     void printQueue() {
61         this.printQueue(0, data.length);
62         // System.out.print("|");
63         // for (int i = 0; i < data.length; i++) {
64         //     System.out.printf("%d", data[i]);
65         //     System.out.printf("|");
66         // }
67         // System.out.println();
68     }
69
70     private void printQueue(int num){
71         System.out.print(data[num]);
72     }
73
74     // 状態表示関数
75     void printQueue(int start, int end) {
76         System.out.print("|");
77         for (int i = start; i < end; i++) {
78             printQueue(i);
79             System.out.printf("|");
80         }
81         System.out.println();
82     }
83
84     int QueueVolume() {
85         return queueCount;
86     }
87 }

```

Listing 3: 実行結果

```

1  5個分のキュー生成
2  1個目のスタックです
3  10個分のキュー生成
4  2個目のスタックです
5  5個分のキュー生成
6  |10|0|0|0|0|
7  |10|20|0|0|0|
8  |10|20|30|0|0|
9  |10|20|30|40|0|
10 |10|20|30|40|50|
11 queue overflow
12 |10|20|30|40|50|
13 10

```

14	50
15	20 30 40 50 0
16	20
17	30 40 50 0 0
18	30
19	40 50 0 0 0
20	40
21	50 0 0 0 0
22	50
23	0 0 0 0 0
24	-1
25	0 0 0 0 0
