



```
1 Resource res1 = new Resource();
2 Resource res2 = new Resource();
3 Task task1 = new Task(
4     35,
5     35,
6     List.of(
7         new Chunk(
8             1,
9             new UniformSampler(new BigDecimal(2), new BigDecimal(2))),
10        new Chunk(
11            2,
12            new UniformSampler(new BigDecimal(1), new BigDecimal(1.5)),
13            List.of(res1)),
14        new Chunk(
15            3,
16            new UniformSampler(new BigDecimal(0.5), new BigDecimal(1))));
17 Task task2 = new Task(
18     50,
19     50,
20     List.of(
21         new Chunk(
22             1,
23             new UniformSampler(new BigDecimal(3), new BigDecimal(4))),
24         new Chunk(
25             2,
26             new UniformSampler(new BigDecimal(3), new BigDecimal(3.5)),
27             List.of(res2)),
28         new Chunk(
29             3,
30             new UniformSampler(new BigDecimal(3), new BigDecimal(3.5))));
31 Task task3 = new Task(
32     80,
33     80,
34     List.of(
35         new Chunk(
36             1,
37             new UniformSampler(new BigDecimal(4), new BigDecimal(5))),
38         new Chunk(
39             2,
40             new UniformSampler(new BigDecimal(4), new BigDecimal(4.5))));
41 Task task4 = new Task(
42     110,
43     110,
44     List.of(
45         new Chunk(
46             1,
47             new UniformSampler(new BigDecimal(5), new BigDecimal(5.5))),
48         new Chunk(
49             2,
50             new UniformSampler(new BigDecimal(2), new BigDecimal(3)),
51             List.of(res2))));
52 Task task5 = new Task(
53     160,
54     160,
55     List.of(
56         new Chunk(
57             2,
58             new UniformSampler(new BigDecimal(3.5), new BigDecimal(4)),
59             List.of(res1))));
60 TaskSet taskSet = new TaskSet(Set.of(task1, task2, task3, task4, task5));
61 ResourcesProtocol pcp = new PriorityCeilingProtocolFaultAcquireResource(0.2);
62 Scheduler rm = new RMScheduler(taskSet, pcp, 1500);
63 rm.scheduleDataset(10);
```