

pseudo1.cpp

```
1
2  enum SERVICE
3  {
4      DELIVERY,
5      CONSULT,
6      _TOT_SERVICES
7  };
8
9  class SantaClausV1 = Monitor
10 {
11     int TOT[_TOT_SERVICES];
12     int MIN_ELVES;
13
14     condition_variable await_someone;
15     condition_variable wait_all_passed;
16     condition_variable wait_greetings;
17
18     condition_variable wait_service[_TOT_SERVICES];
19     int turnstile[_TOT_SERVICES];
20     condition_variable wait_end_of_service;
21     bool end_of_service;
22
23     SantaClausV1(int n_reindeer, int n_elves, int min_elves)
24     {
25         TOT[DELIVERY] = n_reindeer;
26         TOT[CONSULT] = n_elves;
27         MIN_ELVES = min_elves;
28         turnstile = {0};
29         end_of_service = false;
30     }
31
32     entry void new_service(SERVICE s)
33     {
34         if (await_someone.any()) // Santa is free
35             await_someone.notify_one();
36         while (turnstile[s] == 0)
37             wait_service[s].wait(lock);
38
39         turnstile[s]--;
40         if (turnstile[s] > 0)
41             wait_service[s].notify_one();
42         else
43             wait_all_passed.notify_one();
44         while (!end_of_service)
45             wait_end_of_service.wait(lock);
46
47         if (wait_end_of_service.any())
48             wait_end_of_service.notify_one();
49         else
50             wait_greetings.notify_one();
51     }
```

```

52
53     entry void start_service(SERVICE& s)
54     {
55         while (!(wait_service[DELIVERY].getCnt() == TOT[DELIVERY] ||
wait_service[CONSULT].getCnt() >= MIN_ELVES))
56             await_someone.wait(lock);
57
58         if (wait_service[DELIVERY].getCnt() == TOT[DELIVERY]) // serving the reindeer
59             s = DELIVERY;
60         else // serving the elves
61             s = CONSULT;
62         turnstile[s] = (s == DELIVERY ? TOT[s] : MIN_ELVES);
63         end_of_service = false;
64         wait_service[s].notify_one(); // first reindeer/elf awakening
65         while (turnstile[s] > 0)
66             wait_all_passed.wait(lock);
67     }
68
69     entry void end_service()
70     {
71         end_of_service = true;
72         wait_end_of_service.notify_one();
73         while (wait_end_of_service.any())
74             wait_greetings.wait(lock);
75     }
76 };
77
78 SantaClausV1 sc = SantaClausV1(9, 10, 3);
79
80 void reindeer(SantaClausV1& sc)
81 {
82     while (true)
83     {
84         <on vacation and wait Christmas>
85         <head back to the North Pole>
86         sc.new_service(DELIVERY);
87         <head back to the Pacific Islands>
88     }
89 }
90
91 void elf(SantaClausV1& sc, int id)
92 {
93     while (true)
94     {
95         <make toys>
96         sc.new_service(CONSULT);
97     }
98 }
99
100 void santa(SantaClausV1& sc)
101 {
102     SERVICE s;
103
104     while (true)

```

```
105     {
106         sc.start_service(s);
107         if(s == DELIVERY)
108             <delivering toys>
109         else
110             <arguing with a bunch of stupid elves>
111         sc.end_service();
112     }
113 }
114
```