pseudo2.cpp

```cpp
enum SERVICE
{
    DELIVERY,
    CONSULT,
    _TOT_SERVICES
};

class SantaClausV2 = Monitor
{
    int TOT[_TOT_SERVICES];
    int MIN_ELVES;

    condition_variable await_someone;
    condition_variable wait_all_passed;
    condition_variable wait_greetings;
    condition_variable wait_reindeer_return;

    condition_variable wait_service[_TOT_SERVICES];
    int turnstile[_TOT_SERVICES];
    condition_variable wait_end_of_service;
    bool end_of_service;

    SantaClausV2(int n_reindeer, int n_elves, int min_elves)
    {
        TOT[DELIVERY] = n_reindeer;
        TOT[CONSULT] = n_elves;
        MIN_ELVES = min_elves;
        turnstile = {0};
        end_of_service = false;
    }

    entry void new_consult()
    {
        if (await_someone.any()) // Santa is free
            await_someone.notify_one();
        while (turnstile[CONSULT] == 0)
            wait_service[CONSULT].wait(lock);

        turnstile[CONSULT]--;
        if (turnstile[CONSULT] > 0)
            wait_service[CONSULT].notify_one();
        else
            wait_all_passed.notify_one();
        while (!end_of_service)
            wait_end_of_service.wait(lock);

        if (wait_end_of_service.any())
            wait_end_of_service.notify_one();
        else
            wait_greetings.notify_one();
```

```cpp
 52          }
 53
 54      entry void new_delivery()
 55      {
 56          if (await_someone.any()) // Santa is free
 57              await_someone.notify_one();
 58          while (turnstile[DELIVERY] == 0)
 59              wait_service[DELIVERY].wait(lock);
 60
 61          turnstile[DELIVERY]--;
 62          if (turnstile[DELIVERY] > 0)
 63              wait_service[DELIVERY].notify_one();
 64          else
 65              wait_all_passed.notify_one();
 66      }
 67
 68      entry void end_delivery()
 69      {
 70          if (wait_end_of_service.getCnt() == TOT[DELIVERY] - 1)
 71              wait_reindeer_return.notify_one();
 72          while (!end_of_service)
 73              wait_end_of_service.wait(lock);
 74
 75          if (wait_end_of_service.any())
 76              wait_end_of_service.notify_one();
 77          else
 78              wait_greetings.notify_one();
 79      }
 80
 81      entry void start_service(SERVICE& s)
 82      {
 83          while (!(wait_service[DELIVERY].getCnt() == TOT[DELIVERY] ||
    wait_service[CONSULT].getCnt() >= MIN_ELVES))
 84              await_someone.wait(lock);
 85
 86          if (wait_service[DELIVERY].getCnt() == TOT[DELIVERY]) // serving the reindeer
 87              s = DELIVERY;
 88          else // serving the elves
 89              s = CONSULT;
 90          turnstile[s] = (s == DELIVERY ? TOT[s] : MIN_ELVES);
 91          end_of_service = false;
 92          wait_service[s].notify_one(); // first reindeer/elf awakening
 93          while (turnstile[s] > 0)
 94              wait_all_passed.wait(lock);
 95      }
 96
 97      entry void end_service(SERVICE s)
 98      {
 99          if (s == DELIVERY)
100              while (wait_end_of_service.getCnt() < TOT[s])
101                  wait_reindeer_return.wait(lock);
102
103          end_of_service = true;
104          wait_end_of_service.notify_one();
```

```cpp
105              while (wait_end_of_service.any())
106                  wait_greetings.wait(lock);
107          }
108      };
109
110      SantaClausV2 sc = SantaClausV2(9, 10, 3);
111
112      void reindeer(SantaClausV2& sc)
113      {
114          while (true)
115          {
116              <on vacation and wait Christmas>
117              <head back to the North Pole>
118              sc.new_delivery();
119              <delivering toys>
120              sc.end_delivery();
121              <head back to the Pacific Islands>
122          }
123      }
124
125      void elf(SantaClausV2& sc)
126      {
127          while (true)
128          {
129              <make toys>
130              sc.new_consult();
131          }
132      }
133
134      void santa(SantaClausV2& sc)
135      {
136          SERVICE s;
137
138          while (true)
139          {
140              sc.start_service(s);
141              if(s == DELIVERY)
142                  <delivering toys>
143              else
144                  <arguing with a bunch of stupid elves>
145              sc.end_service(s);
146          }
147      }
148
```