```
 1.  #include <fstream>
 2.  #include <iostream>
 3.  #include <iomanip>
 4.  #include <string>
 5.  using namespace std;
 6.  #include "Queue04.h"
 7.  class Sim : public Queue {
 8.      int served;
 9.      int totalresponse;
10.      int maxresponse;
11.      void Arrival(int time);
12.      void Departure(int time);
13.      void Print(ostream& o,char* t,int v,char* u);
14.      void Print(ostream& o,char* t,double v,char* u);
15.  public:
16.      Sim();
17.      void Run(istream& i);
18.      void Print(ostream& o);
19.  };
```

Figure 1. Single-server queue simulation **class** Sim

Sim::*Sim*()
1. Initialize member *served* to zero.
2. Initialize member *totalresponse* to zero.
3. Initialize member *maxresponse* to zero.

void *Sim*::*Run*(*istream*& *i*)
1. Declare integer variable *time* and initialize it to zero.
2. **for(;;){**
   - 2.1. Declare integer variable *code*.
   - 2.2. Scan input file stream *i* for the next integer. Store the integer into variable *code*.
   - 2.3. **if (***i***.***eof***()) break;**
   - 2.4. **switch (***code***) {**
     - 2.4.1. **case 1: //Nothing happened**
     - 2.4.2. **case 2: //A customer arrived**
       - 2.4.2.1. Call function *Arrival*(*time*) to process the new arrival.
     - 2.4.3. **case 3: //A customer departed**
       - 2.4.3.1. Call function *Departure*(*time*) to process the departing customer.
     - 2.4.4. **case 4: //A customer arrived and another customer departed**
       - 2.4.4.1. Call function *Arrival*(*time*) to process the new arrival.
       - 2.4.4.2. Call function *Departure*(*time*) to process the departing customer.
     - 2.4.5. **default: //The code is erroneous**
       - 2.4.5.1. **throw** *InvalidCodeException*(*code*)
   - 2.5. Increment the value of variable *time* by 15 seconds.

void *Sim*::*Print*(*ostream*& ***o***)

1. Call private member function Print having arguments, "Customers served", member *served*, and "customers"
2. Call private member function Print having arguments, "Maximum response", member *maxresponse*, and "seconds"
3. Call private member function Print having arguments, "Simulated mean response", member *totalresponse/served*, and "seconds"

void *Sim*::*Arrival*(**int** *t)*

1. **if** the queue is full **return**. This situation is roughly equivalent to the store being full.
2. Put the arrival time *t* on the queue.

void *Sim*::*Departure*(**int** *t)*

1. **if** the queue is empty **return**. This situation is roughly equivalent to the store being empty.
2. Find the integer value *d* equal to the difference between the current time *t* and the value of the time removed from the queue. Variable *d* is an abbreviation for *difference*.
3. **if** *d* is larger than the maximum response time, *maxresponse*, assign the value of *d* to member *maxresponse*.
4. Add *d* to the sum of all response times of customers that were served, member *totalresponse*.
5. Increment the record of the number of customers that were served, member *served*.