

```
1:
2: # test; M/M/1 queue -- exponential ("Markov") job interarrivals,
3: # exponential service times, 1 server
4:
5: mm1 <- function(meaninterarrv, meansrv, timelim, dbg=FALSE) {
6:   # set up structures
7:   simlist <-
8:     newsim(timelim, 3, appcols=c('arrvtime', 'jobnum'), aevntset=TRUE, dbg)
9:   simlist$reactevent <- mmlreact
10:  simlist$arrvrate <- 1 / meaninterarrv
11:  simlist$srbrate <- 1 / meansrv
12:  simlist$totjobs <- 0
13:  simlist$totwait <- 0.0
14:  simlist$queue <- newqueue(simlist)
15:  simlist$srvrbusy <- FALSE
16:  # defining job numbers is good practice, always invaluable during
17:  # debugging
18:  simlist$jobnum <- 0
19:  # event type codes: 1 for arrival, 2 for service completion
20:  simlist$arrvevnt <- 1
21:  simlist$srvevnt <- 2
22:
23:  exparrivals(simlist, meaninterarrv)
24:
25:  # start sim
26:  mainloop(simlist)
27:
28:  # sim done
29:  # should print out something near 1 / (srbrate - arrvrate)
30:  cat("mean wait: ")
31:  print(simlist$totwait / simlist$totjobs)
32: }
33:
34: # what new events are triggered by the occurrence of an old one?
35: mmlreact <- function(evnt, simlist) {
36:   etype <- evnt['evnttype']
37:   if (etype == simlist$arrvevnt) { # job arrival
38:     # start newly-arrived job or queue it
39:     if (!simlist$srvrbusy) { # server free, start job service
40:       simlist$srvrbusy <- TRUE
41:       srvduration <- rexp(1, simlist$srbrate)
42:       schedevnt(simlist, simlist$currttime+srvduration, simlist$srvevnt,
43:         evnt[3:4]) # copy over previous data for this job
44:     } else { # server busy, add job to queue
45:       appendfcfs(simlist$queue, evnt)
46:     }
47:   } else { # etype = simlist$srvevnt, job completion
48:     # bookkeeping
49:     simlist$totjobs <- simlist$totjobs + 1
50:     # wait time = job completion time - job arrival time
51:     simlist$totwait <- simlist$totwait + simlist$currttime - evnt[3]
52:     simlist$srvrbusy <- FALSE
53:     # check queue for waiting jobs
54:     if (nrow(simlist$queue$m) > 0) { # nonempty queue
55:       qhead <- delfcfs(simlist$queue)
56:       # start job service
57:       simlist$srvrbusy <- TRUE
58:       srvduration <- rexp(1, simlist$srbrate)
59:       schedevnt(simlist, simlist$currttime+srvduration, simlist$srvevnt,
60:         qhead[3:4])
61:     }
62:   }
63: }
```