Review Questions and Problems Chapter 8

1. Explain the fundamental problem in *shared data access* using a *producer-consumer* structure as an example. Draw the corresponding activity diagram.

2. Explain the difference between a *binary semaphore* and a *counting semaphore*.

3. What is a *critical section* in the context of concurrent system?

4. Explain the difference between a *semaphore* and a *mutex*.

5. What conditions must hold to create a *deadlock*, and what can you do to avoid deadlocks in a concurrent system?

6. In two *run-to-completion* tasks TA and TB there shall be executed two actions A1, A2 and B1,B2, respectively. The tasks are repeatedly activated from the outside.

   In TA either action A1 or action A2 is executed, the selection being controlled by a random variable ZV. Thereafter the task terminates itself.
   In TB, first B1 is executed, and thereafter B2. Then the task terminates itself.

   A) In the first configuration make sure that A1 and B2 as well as A2 and B1 never execute in parallel. Make use of *semaphores* for this purpose. Do not impose any unnecessary restrictions. Draw the corresponding *activity diagram*. Indicate how each semaphore is to be initialized.

   B) In the second configuration make sure that B1 is only executed after A1 has been executed. Do not impose any unnecessary restrictions. Draw the corresponding *activity diagram*. Indicate how each *semaphore* is to be initialized.

   C) In the third configuration make sure that B1 is only executed after A1 and A2 have been executed. Make use of *event flags* for this purpose. Do not impose any unnecessary restrictions. Draw the corresponding *activity diagram*. Indicate how each *event flag* is to be initialized. Do not forget to reset the event flags appropriately.