1) What is the best action a system can take when deadlock is detected? (2.5 points)

2. When deadlock occurs, assuming that only two program units are causing the deadlock, one of the involved program units should be gracefully terminated, thereby allowed the other to continue.

2) Busy waiting is a method whereby a task waits for a given event by continuously checking for that event to occur. What is the main problem with this approach? (2.5 points)

3. The main problem with busy waiting is that machine cycles are wasted in the process.

3) Suppose two tasks, A and B, must use the shared variable Buf_Size. Task A adds 2 to Buf_Size, and task B subtracts 1 from it. Assume that such arithmetic operations are done by the three-step process of fetching the current value, performing the arithmetic, and putting the new value back. In the absence of competition synchronization, what sequences of events are possible and what values result from these operations? Assume that the initial value of Buf_Size is 6. (5 points)

Sequence 1:  
A fetches the value of BUF_SIZE (6)  
A adds 2 to the value (8)  
A puts 8 in BUF_SIZE  
B fetches the value of BUF_SIZE (8)  
B subtracts 1 (7)  
B put 7 in BUF_SIZE  
BUF_SIZE = 7
Sequence 2:

A fetches the value of $\text{BUF\_SIZE}$ (6)
B fetches the value of $\text{BUF\_SIZE}$ (6)
A adds 2 (8)
B subtracts 1 (5)
A puts 8 in $\text{BUF\_SIZE}$
B puts 5 in $\text{BUF\_SIZE}$
$\text{BUF\_SIZE} = 5$

Sequence 3:

A fetches the value of $\text{BUF\_SIZE}$ (6)
B fetches the value of $\text{BUF\_SIZE}$ (6)
A adds 2 (8)
B subtracts 1 (5)
B puts 5 in $\text{BUF\_SIZE}$
A puts 8 in $\text{BUF\_SIZE}$
$\text{BUF\_SIZE} = 8$