CS 318/418/618, Fall 2022

Homework #3

Ryan Huang huang@cs.jhu.edu

1. Microsoft .NET provides a synchronization primitive called a `CountdownEvent`. Progra tion of many threads ownEvent is initializ two states, nonsigr the nonsignalled state to nen the internal cou nEvent transitions ting threads. Once a to signalled, the Cou nonsignalled state, a to decrease the cou gnalled state, Wait, D mmediately.

   (a) Us nEvent using lo ing methods:

   ```
   cl
           void Increment () { ... }
           void Decrement () { ... }
           void Wait () { ... }
   }
   ```

   Notes:

   - The CountdownEvent constructor takes an integer count as input and initializes the CountdownEvent counter with count. Positive values of count cause the CountdownEvent to be constructed in the nonsignalled state. Other values of count will construct it in the signalled state.

- Increment increments the internal counter.
- Decrement decrements the internal counter. If the counter reaches zero, the CountdownEvent transitions to the signalled state and unblocks any waiting threads.
- Wait blocks the calling thread if the CountdownEvent is in the nonsignalled state, and otherwise returns.
- Each of these methods is relatively short.

(b) Semaphores also increment and decrement. How do the semantics of a CountdownEvent differ from a Semaphore?

2. A comm set of threads do a c , all threads must fi e $i+1$. One way to end of each phase, the number of thre ocks until all of the ads proceed. You ma ach iteration, and tha ame value.

(a) Im revious exercise

(b) W mantics.

```
mo

}
```

(c) Im variable.

```
class Barrier {
  ...private variables...
  void Done (int n) {
    ...
  }
  ...
}
```

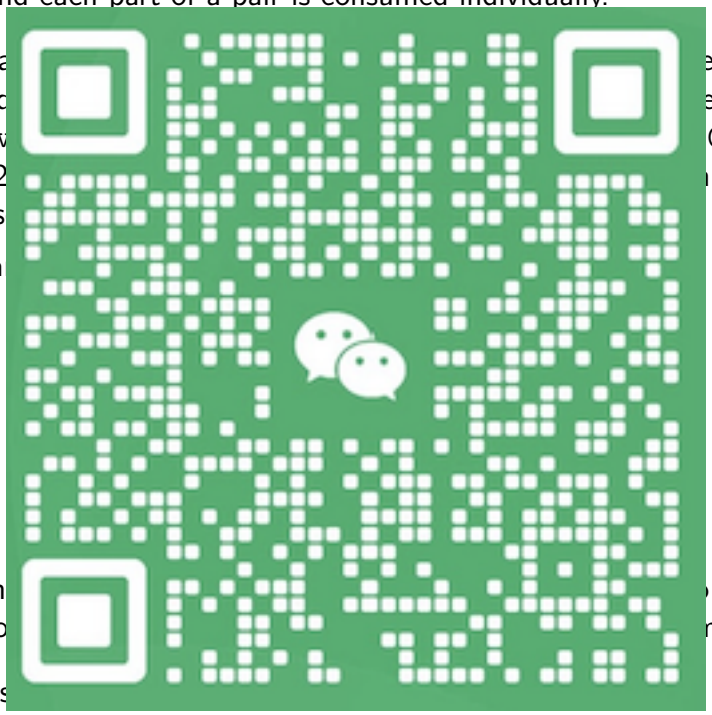
3. Consider a problem in which there is a producer p and two consumers c1 and c2. The producer produces pairs of values $< a, b >$. The producer

does not have to wait in Put for a consumer, and the monitor will have to accumulate the values in auxiliary data structures to ensure nothing gets lost (you can assume the use of lists or arrays). Assume that Put can accumulate at most k pairs of values. Consumer c1 consumes the a values of these pairs and c2 consumes the b values of these pairs. A consumer consumes only one value per call.

> Hint: This problem is very similar to the producer/consumer problem-it just so happens that objects are produced in pairs, and each part of a pair is consumed individually.

Write a e three entry
metho e values, `int`
`GetA(v GetB(void)`
that c2 , you should
only us

An exa



4. Demon they can be
used to ns.

5. [Anders modelling of
oceans. The inner loop of the program matches atoms of different types as they form molecules. In an excessive reliance on threads, each atom is represented by a thread.

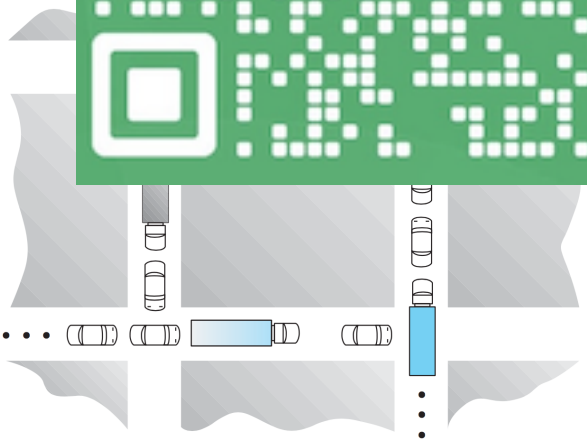
   (a) Your task is to write code to form water out of two hydrogen threads and one oxygen thread (`H2O`). You are to write the two procedures: `HArrives()` and `OArrives()`. A water molecule forms when two `H` threads are present and one `O` thread; otherwise, the atoms must wait. Once all three are present, one of the threads calls `MakeWater()`, and only then, all three depart.

(b) The company wants to extend its work to handle cloud modelling. Your task is to write code to form ozone out of three oxygen threads. Each of the threads calls `OArrives()`, and when three are present, one calls `MakeOzone()`, and only then, all three depart.

(c) Extending the product line into beer production, your task is to write code to form alcohol (`C2H6O`) out of two carbon atoms, six hydrogens, and one oxygen. You must use locks and Mesa-style condition variables to implement your solutions. Obviously, an atom that arrives after the molecule is made must wait for a different group of at d you should co e no useless wa mber of each ty

6. [Silbers nchronization tool ca st implementations , or perhaps order w r locks favor neither ds in a FIFO queue. on tool.

7. [Silbers owing figure.



a) Show that the four necessary conditions for deadlock indeed hold in this example.

b) State a simple rule that will avoid deadlocks in this system

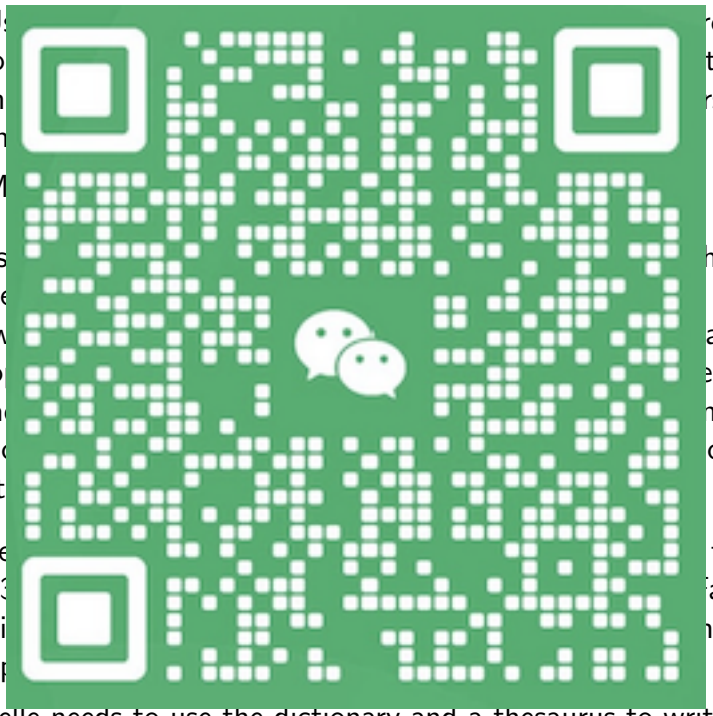
8. [Silberschatz] A single-lane bridge connects the two Vermont villages of North Tunbridge and South Tunbridge. Farmers in the two villages use this bridge to deliver their produce to the neighboring town. The bridge can become deadlocked if a northbound and a southbound farmer get on the bridge at the same time. (Vermont farmers are stubborn and are unable to back up.) Using semaphores and/or mutex locks, design an algorithm in pseudocode that prevents deadlock.

   (a) U revents dead-
   lo the situation
   in rs from using
   th

   (b) M

9. [Silbers hers problem
   (See Se of the prob-
   lem), w able and any
   philoso ests for forks
   are ma ning whether
   a parti ck given the
   current

10. Annabe term papers
    in CS 3 avorite Race
    Conditi e dictionary,
    two cop



Annabelle needs to use the dictionary and a thesaurus to write her paper;

- Bertrand needs a thesaurus and a coffee cup to write his paper;
- Chloe needs a dictionary and a thesaurus to write her paper;
- Dag needs two coffee cups to write his paper (he likes to have a cup of regular and a cup of decaf at the same time to keep himself in balance).

Consider the following state:

- Annabelle has a thesaurus and need the dictionary.

- Bertrand has a thesaurus and a coffee cup.
- Chloe has the dictionary and needs a thesaurus.
- Dag has a coffee cup and needs another coffee cup.
  - Is the system deadlocked in this state? Explain using a resource allocation graph.
  - Is this state reachable if the four people allocated and released their resources using the Banker's algorithm? Explain.