LE19.2

Bookmark this page

LE19.2.1: Mutual Exclusion

0.0/1.0 point (ungraded)

You are experimenting with a pair of simple processes, A and B, that share an integer variable X as well as a semaphore S:

```
Process A

int Y;
wait(S);
Y = X + 1;
X = Y;
signal(S)
```

```
Shared Memory

int X = 1;
Semaphore S = 1;
```

```
Process B

int Z;
wait(S);
Z = X * 5;
X = Z;
signal(S);
```

The processes run on a timeshared system whose scheduler promises no constraints on process scheduling other than those imposed by the use of semaphores by the processes. Note that each process uses a local variable, in addition to the shared variable X. The two processes are run simultaneously; execution of their statements may be interleaved. You may assume that the execution of each line of code is atomic (that is, un-interrupted by the scheduler). Note that the initial values of X and S are both 1.

1. After execution of the above pair of processes, what final values of the variable X are possible? Select all possible values. Select “None” if one or both processes will never run to completion.

Possible final values of X, or “None”:

- [ ] 0
- [ ] 1
- [ ] 2
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6
- [ ] 7
- [ ] 8
- [ ] 9
- [ ] 10
- [ ] None

2. Suppose the initial value of the semaphore in the above code is changed to 2. What final values are now possible for X?

Possible final values of X, or “None”:

- [ ] 0
- [ ] 1

Calculator

- [ ] 2
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6
- [ ] 7
- [ ] 8
- [ ] 9
- [ ] 10
- [ ] None

Finally, you use 0 as the initial value of S, and delete two lines of code, resulting in:

```
  Process A

  int Y;
  wait(S);
  Y = X + 1;
  X = Y;
  // Line
removed
```

```
  Shared Memory

  int X = 1;
  Semaphore S = 0;  // Semaphore initialized to
0
```

```
  Process B

  int Z;
  // Line
removed
  Z = X * 5;
  X = Z;
  signal(S);
```

3. After execution of the above pair of processes, what final values of the variable X are possible? Select all possible values. Select "None" if one or both processes will never run to completion.

Possible final values of X, or "None":

- [ ] 0
- [ ] 1
- [ ] 2
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6
- [ ] 7
- [ ] 8
- [ ] 9

Calculator

- [ ] 10
- [ ] None

Submit

LE19.2.2: Using Semaphores

0.0/1.0 point (ungraded)

In lecture, you saw the use of semaphores to mediate a communication stream between a Producer and a Consumer process. In this problem, we assume the existence of three asynchronous processes: a Producer process, producing a stream of characters; a Consumer process, which consumes a stream of characters; and a Filter process spliced between the Consumer and Producer processes. The Filter process takes characters from the producer, processes them (via a **translate** function), and passes the result to the Consumer process. The Producer and Consumer processes each communicate directly only with the Filter process.

The following is in Shared Memory (shared among Producer, Filter, and Consumer processes):

```
Semaphore charsA=???, spaceA=???, charsB=???, spaceB=???;
char buf[100];
char indata;
int in=0, out=0;
```

and the following code runs in the Filter Process:

```
while (1) {                  /* loop forever… */
   char temp;                /* local variable */

   wait(charsA);
   temp = indata;
   signal(spaceA);
   temp = translate(temp);   /* do the actual translation */
   wait(spaceB);
   buf[in] = temp;
   in = (in+1)%100;          /* increment ‘in’ modulo 100 */
   signal(charsB);
}
```

1. What is the maximum number of characters that can be produced by the Producer process but not yet processed by the Filter process?

   Maximum unprocessed characters produced:

2. What are appropriate initial values for each of the semaphores?

   Initial value for charsA:

   Initial value for spaceA:

   Initial value for charsB:

   Initial value for spaceB:

Submit

Calculator

Discussion

Hide Discussion

Topic: 19. Concurrency and Synchronization / LE19.2

Add a Post

Show all posts by recent activity

[STAFF] Setting up LE19.2.2, Producer/Filter/Consumer

Community TA

10

Previous Next

© All Rights Reserved

© 2021 edX Inc. All rights reserved.
深圳市恒宇博科技有限公司 粤ICP备17044299号-2

Calculator