

ICS'13 Homework 10

Problem 1: Concurrency basic

Starvation (in relation to threads) refers to:

- (a) A thread waiting for a lock indefinitely.
- (b) A semaphore that gets locked but the thread never unlocks it after use.
- (c) A thread is spawned but never joins the main thread when finished.
- (d) A process fails to spawn a new thread because it's hit the maximum number of threads allowed.

Problem 2: Producer-consumer problem

Let P denote the number of producers, C the number of consumers, and N the buffer size in units of items. For each of the following scenarios, indicate whether the mutex semaphore in `sbuf_insert` and `sbuf_remove` is necessary or not.

A. $P=1, C=1, N>1$

B. $P=1, C=1, N=1$

C. $P>1, C>1, N=1$

Problem 3: Dead Lock

Can the following program deadlock? Why or why not?

Initially: $a=1, b=1, c=1$.

Thread 1: Thread 2:

$P(a);$ $P(c);$

$P(b);$ $P(b);$

$V(b);$ $V(b);$

$P(c);$ $V(c);$

$V(c);$

$V(a);$

Problem 4: Race

Does the following program contain a race on the value of `myid`?

```
sem_t s; /* semaphore s */
void *foo(void *vargp) {
    int myid;
    myid = *((int *)vargp);
    V(&s);
    printf("Thread %d\n", myid);
}
int main() {
    pthread_t tid[2];
    int i;
    sem_init(&s, 0, 0); /* S=0 INITIALLY */
```

```
for (i = 0; i < 2; i++) {  
    Pthread_create(&tid[i], 0, foo, &i);  
    P(&s);  
}  
Pthread_join(tid[0], 0);  
Pthread_join(tid[1], 0);  
}
```

Problem 5: Virtual Memory (Open Question)

What are the advantages of using virtual memory instead of physical memory?