

Operating Systems Lab

Classical Problems

Aadhitya Swarnesh



— 28Feb2020

Q1) Producer-Consumer

```
#include<stdio.h>
#include<pthread.h>
#include<sys/types.h>
#include<unistd.h>
#include<stdlib.h>
#include<semaphore.h>

sem_t empty,full,mutex;
char buf[10];

void* thread_fun1(void* arg)
{
    int i;
    printf("inside producer\n");
    for(i=0;i<10;i++)
    {
        sem_wait(&empty);
        sem_wait(&mutex);
        buf[i]=i;
        printf("item produced is %d\n",buf[i]);
        sem_post(&mutex);
        sem_post(&full);
        sleep(2);
    }
    pthread_exit("producer\n");
}

void * thread_fun2(void* arg)
{
    int j;
    printf("inside consumer\n");
    for(j=0;j<10;j++)
    {
        sem_wait(&full);
        sem_wait(&mutex);
        j=buf[j];
        printf("consumed item is:%d\n",buf[j]);
        sem_post(&mutex);
        sem_post(&empty);
        sleep(3);
    }
    pthread_exit("consumer\n");
}
```

```

}

int main()
{
    pthread_t pid1, pid2;
    sem_init(&empty, 0, 10);
    sem_init(&full, 0, 0);
    sem_init(&mutex, 0, 1);
    void *status;
    pthread_create(&pid1, NULL, thread_fun1, NULL);
    pthread_create(&pid2, NULL, thread_fun2, NULL);
    pthread_join(pid1, &status);
    //printf("the exited status of 1st is %s\n", (char*)status);
    pthread_join(pid2, &status);
    //printf("the exited status of %s\n", (char*)status);
    return 0;
}

```

```

(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$ gcc q1.c
q1.c:50:5: warning: 'sem_init' is deprecated [-Wdeprecated-declarations]
    sem_init(&empty, 0, 10);
    ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/semaphore.h:55:42: note: 'sem_init' has been explicitly marked deprecated here
int sem_init(sem_t *, int, unsigned int) __deprecated;
                                         ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/cdefs.h:187:40: note: expanded from macro '__deprecated'
#define __deprecated      __attribute__((__deprecated__))
                                         ^
q1.c:51:5: warning: 'sem_init' is deprecated [-Wdeprecated-declarations]
    sem_init(&full, 0, 0);
    ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/semaphore.h:55:42: note: 'sem_init' has been explicitly marked deprecated here
int sem_init(sem_t *, int, unsigned int) __deprecated;
                                         ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/cdefs.h:187:40: note: expanded from macro '__deprecated'
#define __deprecated      __attribute__((__deprecated__))
                                         ^
q1.c:52:5: warning: 'sem_init' is deprecated [-Wdeprecated-declarations]
    sem_init(&mutex, 0, 1);
    ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/semaphore.h:55:42: note: 'sem_init' has been explicitly marked deprecated here
int sem_init(sem_t *, int, unsigned int) __deprecated;
                                         ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/cdefs.h:187:40: note: expanded from macro '__deprecated'
#define __deprecated      __attribute__((__deprecated__))
                                         ^
3 warnings generated.
(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$ ./a.out
inside producer
inside consumer
consumed item is:0
item produced is 0
item produced is 1
consumed item is:1
item produced is 2
consumed item is:2
item produced is 3
item produced is 4
consumed item is:3
item produced is 5
consumed item is:4
item produced is 6
item produced is 7
consumed item is:5
item produced is 8
consumed item is:6
item produced is 9
consumed item is:7
consumed item is:8
consumed item is:9
(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$ 

```

Q2) Dining Philosopher

```
#include <pthread.h>
#include <semaphore.h>
#include <stdio.h>

#define N 5
#define THINKING 2
#define HUNGRY 1
#define EATING 0
#define LEFT (phnum + 4) % N
#define RIGHT (phnum + 1) % N

int state[N];
int phil[N] = { 0, 1, 2, 3, 4 };

sem_t mutex;
sem_t S[N];

void test(int phnum)
{
    if (state[phnum] == HUNGRY
        && state[LEFT] != EATING
        && state[RIGHT] != EATING) {
        // state that eating
        state[phnum] = EATING;

        sleep(2);

        printf("Philosopher %d takes fork %d and %d\n",
               phnum + 1, LEFT + 1, phnum + 1);

        printf("Philosopher %d is Eating\n", phnum + 1);

        // sem_post(&S[phnum]) has no effect
        // during takefork
        // used to wake up hungry philosophers
        // during putfork
        sem_post(&S[phnum]);
    }
}

// take up chopsticks
void take_fork(int phnum)
{
    sem_wait(&mutex);

    // state that hungry
    state[phnum] = HUNGRY;

    printf("Philosopher %d is Hungry\n", phnum + 1);

    // eat if neighbours are not eating
    test(phnum);

    sem_post(&mutex);

    // if unable to eat wait to be signalled
    sem_wait(&S[phnum]);
}
```



```

    printf("Philosopher %d is thinking\n", i + 1);
}

for (i = 0; i < N; i++)

    pthread_join(thread_id[i], NULL);
}

```

```

(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$ gcc q2.c
q2.c:26:9: warning: implicit declaration of function 'sleep' is invalid in C99 [-Wimplicit-function-declaration]
    sleep(2);
    ^
q2.c:60:5: warning: implicit declaration of function 'sleep' is invalid in C99 [-Wimplicit-function-declaration]
    sleep(1);
    ^
q2.c:89:9: warning: implicit declaration of function 'sleep' is invalid in C99 [-Wimplicit-function-declaration]
    sleep(1);
    ^
q2.c:99:1: warning: control reaches end of non-void function [-Wreturn-type]
}
^
q2.c:108:5: warning: 'sem_init' is deprecated [-Wdeprecated-declarations]
    sem_init(&mutex, 0, 1);
    ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/semaphore.h:55:42: note: 'sem_init' has been explicitly marked deprecated here
int sem_init(sem_t *, int, unsigned int) __deprecated;
                                         ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/cdefs.h:187:40: note: expanded from macro '__deprecated'
#define __deprecated      __attribute__((__deprecated__))
                                         ^
q2.c:112:9: warning: 'sem_init' is deprecated [-Wdeprecated-declarations]
    sem_init(&S[i], 0, 0);
    ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/semaphore.h:55:42: note: 'sem_init' has been explicitly marked deprecated here
int sem_init(sem_t *, int, unsigned int) __deprecated;
                                         ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/cdefs.h:187:40: note: expanded from macro '__deprecated'
#define __deprecated      __attribute__((__deprecated__))
                                         ^
6 warnings generated.
(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$ ./a.out
Philosopher 1 is thinking
Philosopher 2 is thinking
Philosopher 3 is thinking
Philosopher 4 is thinking
Philosopher 5 is thinking
Philosopher 3 is Hungry
Philosopher 2 is Hungry
Philosopher 1 is Hungry
Philosopher 4 is Hungry
Philosopher 5 is Hungry
Philosopher 2 putting fork 1 and 2 down
Philosopher 2 is thinking
Philosopher 5 putting fork 4 and 5 down
Philosopher 5 is thinking
Philosopher 4 putting fork 3 and 4 down
Philosopher 4 is thinking
Philosopher 3 takes fork 2 and 3
Philosopher 3 is Eating
Philosopher 1 takes fork 5 and 1
Philosopher 1 is Eating
Philosopher 2 is Hungry
Philosopher 5 is Hungry
Philosopher 4 is Hungry

```

```

Philosopher 1 putting fork 5 and 1 down
Philosopher 3 putting fork 2 and 3 down
Philosopher 3 is thinking
Philosopher 4 putting fork 3 and 4 down
Philosopher 4 is thinking
Philosopher 2 putting fork 1 and 2 down
Philosopher 2 is thinking
Philosopher 5 putting fork 4 and 5 down
Philosopher 5 is thinking
Philosopher 1 is thinking
Philosopher 3 is Hungry
Philosopher 1 is Hungry
Philosopher 3 takes fork 2 and 3
Philosopher 3 is Eating
Philosopher 1 takes fork 5 and 1
Philosopher 1 is Eating
Philosopher 3 putting fork 2 and 3 down
Philosopher 3 is thinking
Philosopher 1 putting fork 5 and 1 down
Philosopher 1 is thinking
(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$

```

Q3) Reader Writer Problem :

```
#include<semaphore.h>
#include<stdio.h>
#include<pthread.h>

void *reader(void *);
void *writer(void *);

int readcount=0,writecount=0,sh_var=5,bsize[5];
sem_t x,y,z,rsem,wsem;
pthread_t r[3],w[2];

void *reader(void *i)
{
    printf("\n\n reader- %d is reading", i);
    sem_wait(&z);
    sem_wait(&rsem);
    sem_wait(&x);
    readcount++;
    if(readcount==1)
        sem_wait(&wsem);
    sem_post(&x);
    sem_post(&rsem);
    sem_post(&z);
    printf("\nupdated value : %d ", sh_var);
    sem_wait(&x);
    readcount--;
    if(readcount==0)
        sem_post(&wsem);
    sem_post(&x);
}

void *writer(void *i)
{
    printf("\n\n writer- %d is writing", i);
    sem_wait(&y);
    writecount++;
    if(writecount==1)
        sem_wait(&rsem);
    sem_post(&y);
    sem_wait(&wsem);

    sh_var=sh_var+5;
    sem_post(&wsem);
    sem_wait(&y);
    writecount--;
    if(writecount==0)
        sem_post(&rsem);
    sem_post(&y);
}

int main()
{
    sem_init(&x,0,1);
    sem_init(&wsem,0,1);
    sem_init(&y,0,1);
    sem_init(&z,0,1);
    sem_init(&rsem,0,1);
```

```
pthread_create(&r[0],NULL,(void *)reader,(void *)0);
pthread_create(&w[0],NULL,(void *)writer,(void *)0);
pthread_create(&r[1],NULL,(void *)reader,(void *)1);
pthread_create(&r[2],NULL,(void *)reader,(void *)2);
pthread_create(&r[3],NULL,(void *)reader,(void *)3);
pthread_create(&w[1],NULL,(void *)writer,(void *)3);
pthread_create(&r[4],NULL,(void *)reader,(void *)4);
```

```
pthread_join(r[0],NULL);
pthread_join(w[0],NULL);
pthread_join(r[1],NULL);
pthread_join(r[2],NULL);
pthread_join(r[3],NULL);
pthread_join(w[1],NULL);
pthread_join(r[4],NULL);
```

```
return(0);
```

```
}
```

```
(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$ ./a.out
```

```
reader- 0 is reading
```

```
reader- 1 is reading
```

```
writer- 0 is writing
updated value : 5
```

```
reader- 2 is reading
updated value : 10
updated value : 10
```

```
writer- 3 is writing
```

```
reader- 4 is reading
```

```
reader- 3 is reading
updated value : 15
```

```
(base) Aadhityas-MacBook-Air:28Feb2020 aadhitya$
```