

1. Run this code using any IDE multiple times and analyse the outputs.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pthread.h>
#include <semaphore.h>
int t_id[]={1,2};
void *t_func1(int *id);
void *t_func2(int *id);
int sum=15;
pthread_mutex_t m;
sem_t s;
int main(){
    pthread_t t[2];
    sem_init(&s,0,0);
    pthread_mutex_init(&m,NULL);
    pthread_create(&t[0],NULL,(void *)t_func1,&t_id[0]);
    pthread_create(&t[1],NULL,(void *)t_func2,&t_id[1]);
    for(int i=0;i<2;i++){
        pthread_join(t[i],NULL);
    }
    sem_destroy(&s);
    pthread_mutex_destroy(&m);
    printf("Total sum: %d\n",sum);
    return 0;
}
void *t_func1(int *id){
    sem_wait(&s);
    pthread_mutex_lock(&m);
    for(int i=0;i<5;i++){
        printf("Sum: %d\n",sum);
        sum-=10;
    }
    pthread_mutex_unlock(&m);
    sem_post(&s);
}
void *t_func2(int *id){
    pthread_mutex_lock(&m);
    for(int i=0;i<5;i++){
        printf("Sum: %d\n",sum);
        sum*=3;
    }
    pthread_mutex_unlock(&m);
    sem_post(&s);
}
```

2. Run this code using any IDE multiple times and analyse the outputs.

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

int t_id[]={1,2};
void *t_func1(int *id);
void *t_func2(int *id);
int sum=0;
sem_t s1,s2;

int main(){
    pthread_t t[2];
    sem_init(&s1,0,1);
    sem_init(&s2,0,0);
    pthread_create(&t[0],NULL,(void *)t_func1,&t_id[0]);
    pthread_create(&t[1],NULL,(void *)t_func2,&t_id[1]);
    for(int i=0;i<2;i++){
        pthread_join(t[i],NULL);
    }
    sem_destroy(&s1);
    sem_destroy(&s2);
    printf("Total sum: %d\n",sum);
    return 0;
}

void *t_func1(int *id){
    sem_wait(&s1);
    for(int i=0;i<10;i++){
        printf("Sum: %d\n",sum);
        sum+=10;
    }
    sem_post(&s1);
    sem_post(&s2);
}

void *t_func2(int *id){
    sem_wait(&s2);
    for(int i=0;i<10;i++){
        printf("Sum: %d\n",sum);
        sum-=5;
    }
    sem_post(&s2);
}
```

3. Run this code using any IDE multiple times and analyse the outputs.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pthread.h>
#include <semaphore.h>
int t_id[]={1,2,3};
void *t_func1(int *id);
void *t_func2(int *id);
void *t_func3(int *id);
int sum=13;
sem_t s1,s2,s3;
int main(){
    pthread_t t[3];
    sem_init(&s1,0,0);
    sem_init(&s2,0,1);
    sem_init(&s3,0,0);
    pthread_create(&t[0],NULL,(void *)t_func1,&t_id[0]);
    pthread_create(&t[1],NULL,(void *)t_func2,&t_id[1]);
    pthread_create(&t[2],NULL,(void *)t_func3,&t_id[2]);
    for(int i=0;i<3;i++){
        pthread_join(t[i],NULL);
    }
    sem_destroy(&s1);
    sem_destroy(&s2);
    sem_destroy(&s3);
    printf("Total sum: %d\n",sum);
    return 0;
}
void *t_func1(int *id){
    sem_wait(&s1);
    for(int i=0;i<5;i++){
        printf("Sum: %d\n",sum);
        sum*=2;
    }
    sem_post(&s1);
}
void *t_func2(int *id){
    sem_wait(&s2);
    for(int i=0;i<5;i++){
        printf("Sum: %d\n",sum);
        sum+=7;
    }
    sem_post(&s2);
    sem_post(&s3);
}
void *t_func3(int *id){
    sem_wait(&s3);
    for(int i=0;i<5;i++){
        printf("Sum: %d\n",sum);
        sum-=3;
    }
    sem_post(&s3);
    sem_post(&s1);
}
```