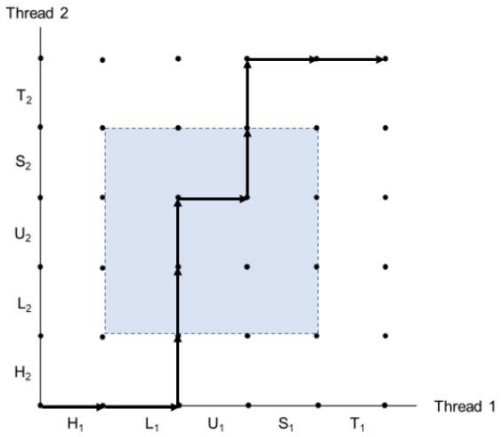
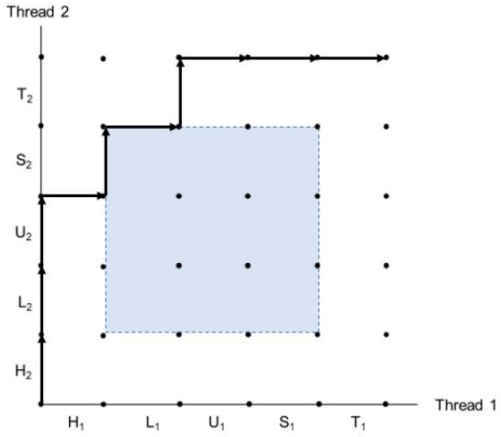
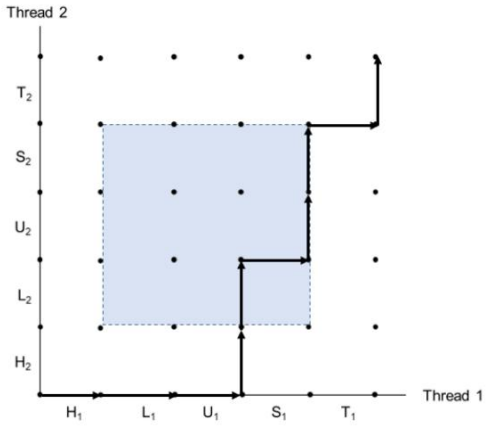
Homework 11

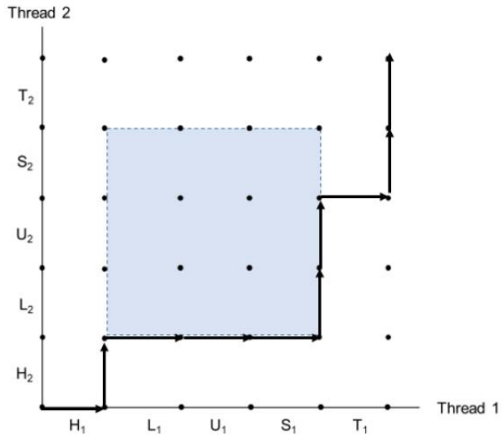
Problem 1





A: cnt=1 B: cnt=2





C: cnt=1 D: cnt=2

Problem 2

1

```
int main() {
    pthread_t tid[N];
    int i, *ptr;

    for (i = 0; i < N; i++) {
        ptr = Malloc(sizeof(int));
        *ptr = i;
        // Create a thread running thread with argument
           ptr
        // Your code here
```

```
            Pthread_create(&tid[i], NULL,
            thread, ptr); Free(ptr);
        }
        // Join all threads
        // Your code here
        for (i = 0; i < N; i++)
            Pthread_join(tid[i], NULL);
    }
```

2. Yes. If the free call executed before the newly created thread, then there will be a segmentation fault caused by accessing a freed pointer.

Problem 3

Swap1: not thread-safe, not reentrant
Swap2: thread-safe, not reentrant
Swap3: thread-safe, reentrant

Problem 4

Yes. If the program enters the deadlock area, then a deadlock will happen.