

# 三峡大坝问题(典型 IPC 问题讨论)

---

1801210840 姜慧强

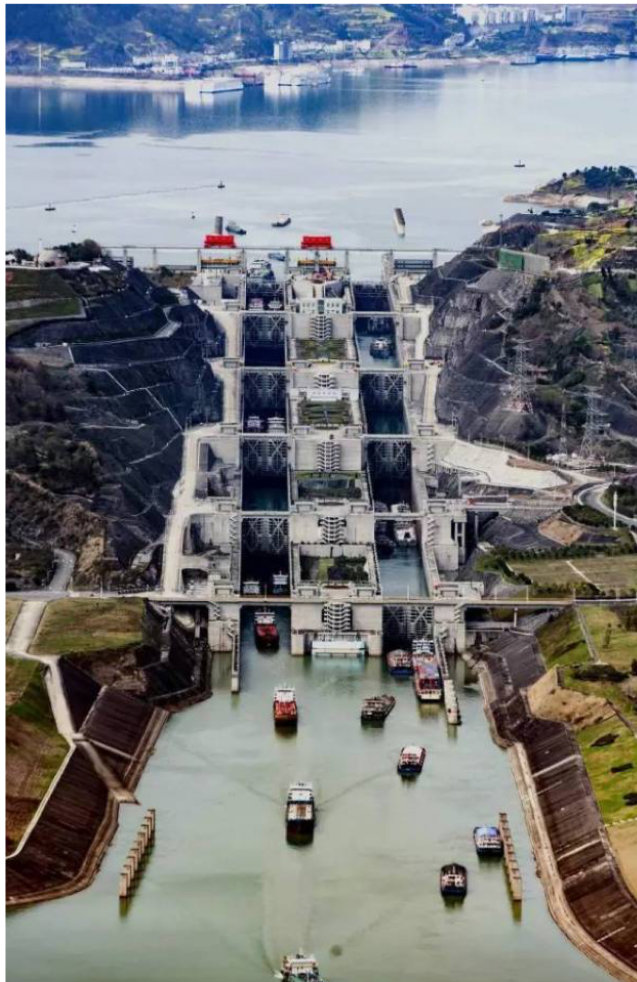
## 问题描述

在三峡大坝上下游各有若干船舶准备借助三峡大坝的船闸来渡过三峡。

三峡大坝 海拔相差 100 余米，通过五级船闸来实现调度。

每个船闸有两个状态(上行/下行)，状态之间相互互斥。

船舶需要依次通过五级船闸才能顺利度过三峡大坝，



## 问题关键点

在保证不发生死锁的情况下尽量提高运行效率，以及尽量避免饥饿的出现。

该问题中出现了以下的互斥关系：

- 每个船闸的进出是互斥的
- 船闸两个状态之间是互斥的

## 解法 PV

### 构造

- 五级船闸各一个信号量, sl0, sl1, sl2, sl3, sl4

- 向上向下开始的各一个信号量保证队首唯一, upBegin, downBegin
- 决定向上还是向下, mutex
- 设置 while 判断对方方向有没有来船, 若有则主动让出 MUTEX

```

/*
 * @Author: gunjianpan
 * @Date: 2019-04-21 16:36:14
 * @Last Modified by: gunjianpan
 * @Last Modified time: 2019-04-21 23:03:03
 */
#include <pthread.h>
#include <semaphore.h>
#include <stdio.h>
#include <stdlib.h>
#define THREADNUM 100
#define LOCKNUM 5
int upNum = 0, downNum = 0;
sem_t *sl[LOCKNUM], *mutex, *upBegin, *downBegin;
pthread_t tidList[THREADNUM];

void *Up(void *b) {
    // up & down sent one ship once
    sem_wait(upBegin);
    if (!upNum) sem_wait(mutex); // flow one direction(no care downNum)
    while (downNum) {
        sem_post(mutex);
        sem_wait(mutex);
    }
    ++upNum;
    pthread_t tid = pthread_self();
    printf("Tid: %p \033[91mBegin ΔΔΔ>>>UP>>>ΔΔΔ\033[0m upNum %d\n", tid, upNum, downNum);

    for (int i = 0; i < LOCKNUM; ++i) {
        sem_wait(sl[i]);
        printf("Tid: %p \033[91m%d\033[0m Up\n", tid, i + 1);
        sem_post(sl[i]);
        if (!i) sem_post(upBegin);
    }
    --upNum;
    printf("Tid: %p \033[91mship C.I.F. >>>Uppppp>>>\033[0m\n", tid);
    if (!upNum) sem_post(mutex);
    return NULL;
}

void *Down(void *a) {
    // up & down sent one ship once
    sem_wait(downBegin);
    if (!downNum) sem_wait(mutex); // flow one direction(no care upNum)
    while (upNum) {
        sem_post(mutex);
        sem_wait(mutex);
    }
    ++downNum;
    pthread_t tid = pthread_self();
    printf("Tid: %p \033[93mBegin ↓↓↓<<<Down<<<↓↓↓\033[0m upNum %d\n", tid, upNum, downNum);
}

```

```

for (int i = LOCKNUM - 1; i >= 0; --i) {
    sem_wait(sl[i]);
    printf("Tid: %p \033[93m%d\033[0m Down\n", tid, i + 1);
    sem_post(sl[i]);
    if (i == LOCKNUM - 1) sem_post(downBegin);
}
--downNum;
printf("Tid: %p \033[93mship C.I.F. <<<<Downnnnnn<<<<\033[0m\n",
tid);
if (!downNum) sem_post(mutex);
return NULL;
}

void thread_one_create(pthread_t nowTid, int types) {
    if (types)
        pthread_create(&nowTid, NULL, Up, NULL);
    else
        pthread_create(&nowTid, NULL, Down, NULL);
}

void thread_one_join(pthread_t nowTid) { pthread_join(nowTid, NULL); }

int main(int argc, char *argv[]) {
    printf("\033[01;34m Write by Jiang Huiqiang 1801210840 in 2019-04-21\n");
    sem_unlink("sem");
    for (int i = 0; i < LOCKNUM; ++i) sl[i] = sem_open("sem", O_CREAT,
0, 1);
    upBegin = sem_open("sem", O_CREAT, 0, 1);
    downBegin = sem_open("sem", O_CREAT, 0, 1);
    mutex = sem_open("sem", O_CREAT, 0, 1);

    for (int i = 0; i < THREADNUM; ++i) thread_one_create(tidList[i], i
% 2);
    for (int i = 0; i < THREADNUM; ++i) thread_one_join(tidList[i]);

    for (int i = 0; i < LOCKNUM; ++i) sem_destroy(sl[i]);

    pthread_exit(NULL);
}

```

```
2. gunjianpan@Mr-J-MacBook-Pro: ~ (zsh)
→ /Users/gunjianpan/Desktop/course/19/os/discuss/MegaDam 10:11:39 PM
Write by Jiang Huiqiang 1801210840 in 2019-04-21
Tid: 0x7000053b7000 Begin ↓↓↓<<<Down<<<↑↑↑ upNum 0 downNum: 1
Tid: 0x7000053b7000 5 Down
Tid: 0x7000054bd000 Begin ↓↓↓<<<Down<<<↑↑↑ upNum 0 downNum: 2
Tid: 0x7000054bd000 5 Down
Tid: 0x7000054bd000 4 Down
Tid: 0x7000054bd000 3 Down
Tid: 0x7000053b7000 4 Down
Tid: 0x7000054bd000 2 Down
Tid: 0x7000055c3000 Begin ↓↓↓<<<Down<<<↑↑↑ upNum 0 downNum: 3
Tid: 0x7000055c3000 5 Down
Tid: 0x7000055c3000 4 Down
Tid: 0x7000055c3000 3 Down
Tid: 0x7000054bd000 1 Down
Tid: 0x7000053b7000 3 Down
Tid: 0x7000053b7000 2 Down
Tid: 0x7000055c3000 2 Down
Tid: 0x7000054bd000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x7000055c3000 1 Down
Tid: 0x7000055c3000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x7000053b7000 1 Down
Tid: 0x7000053b7000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x70000543a000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 1 downNum: 0
Tid: 0x70000543a000 1 Up
Tid: 0x700005540000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 2 downNum: 0
Tid: 0x700005540000 1 Up
Tid: 0x70000543a000 2 Up
Tid: 0x70000543a000 3 Up
Tid: 0x70000543a000 4 Up
Tid: 0x70000543a000 5 Up
Tid: 0x70000543a000 ship C.I.F. >>>>Uppppp>>>>
Tid: 0x700005540000 2 Up
Tid: 0x700005540000 3 Up
Tid: 0x700005540000 4 Up
Tid: 0x700005540000 5 Up
Tid: 0x700005540000 ship C.I.F. >>>>Uppppp>>>>
(base)
22:11:39 in ~ is 📦 vnull on 🐳 v18.09.2 via 🐙 base
→
```

```
2. gunjianpan@Mr-J-MacBook-Pro: ~ (zsh) 10:12:04 PM
Tid: 0x700009da5000 3 Up
Tid: 0x700009da5000 4 Up
Tid: 0x700009da5000 5 Up
Tid: 0x700009da5000 ship C.I.F. >>>>Upppp>>>>
Tid: 0x70000a5d5000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 7 downNum: 0
Tid: 0x70000a6db000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 8 downNum: 0
Tid: 0x70000a7e1000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 9 downNum: 0
Tid: 0x70000a8e7000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 10 downNum: 0
Tid: 0x70000a9ed000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 11 downNum: 0
Tid: 0x70000aaf3000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 12 downNum: 0
Tid: 0x70000abf9000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 13 downNum: 0
Tid: 0x70000acff000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 14 downNum: 0
Tid: 0x70000ae05000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 15 downNum: 0
Tid: 0x70000af0b000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 16 downNum: 0
Tid: 0x70000b011000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 17 downNum: 0
Tid: 0x70000b117000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 18 downNum: 0
Tid: 0x70000b21d000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 19 downNum: 0
Tid: 0x70000b323000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 20 downNum: 0
Tid: 0x70000b429000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 21 downNum: 0
Tid: 0x70000b52f000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 22 downNum: 0
Tid: 0x70000b635000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 23 downNum: 0
Tid: 0x70000b73b000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 24 downNum: 0
Tid: 0x70000b841000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 25 downNum: 0
Tid: 0x70000b947000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 26 downNum: 0
Tid: 0x70000ba4d000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 27 downNum: 0
Tid: 0x70000bb53000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 28 downNum: 0
Tid: 0x70000bc59000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 29 downNum: 0
Tid: 0x70000bd5f000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 30 downNum: 0
Tid: 0x70000be65000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 31 downNum: 0
Tid: 0x70000bf6b000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 32 downNum: 0
Tid: 0x70000c071000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 33 downNum: 0
Tid: 0x70000c177000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 34 downNum: 0
Tid: 0x70000c27d000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 35 downNum: 0
Tid: 0x70000c383000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 36 downNum: 0
Tid: 0x70000c489000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 37 downNum: 0
Tid: 0x70000c58f000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 38 downNum: 0
Tid: 0x70000c695000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 39 downNum: 0
Tid: 0x70000c79b000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 40 downNum: 0
Tid: 0x70000c8a1000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 41 downNum: 0
Tid: 0x70000c9a7000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 42 downNum: 0
Tid: 0x70000caad000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 43 downNum: 0
Tid: 0x70000cbb3000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 44 downNum: 0
Tid: 0x70000ccb9000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 45 downNum: 0
Tid: 0x70000cdbf000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 46 downNum: 0
Tid: 0x70000cec5000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 47 downNum: 0
Tid: 0x70000cfcb000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 48 downNum: 0
Tid: 0x700009fb1000 1 Up
Tid: 0x700009fb1000 2 Up
Tid: 0x700009fb1000 3 Up
Tid: 0x700009fb1000 4 Up
```

FOR HUNGER

设置 totalUpNum, totalDownNum

```

/*
 * @Author: gunjianpan
 * @Date: 2019-04-21 16:36:14
 * @Last Modified by: gunjianpan
 * @Last Modified time: 2019-04-21 22:25:48
 */
#include <pthread.h>
#include <semaphore.h>
#include <stdio.h>
#include <stdlib.h>
#define THREADNUM 100
#define LOCKNUM 5
#define HUNGERNUM 5
int upNum = 0, downNum = 0;
int totalUpNum = 0, totalDownNum = 0;
sem_t *sl[LOCKNUM], *mutex, *upBegin, *downBegin;
pthread_t tidList[THREADNUM];

void *Up(void *b) {
    // up & down sent one ship once
    sem_wait(upBegin);
    if (!upNum) sem_wait(mutex); // flow one direction(no care downNum)
    while (downNum || totalUpNum - totalDownNum > HUNGERNUM) {
        sem_post(mutex);
        sem_wait(mutex);
    }
    ++totalUpNum;
    ++upNum;
    pthread_t tid = pthread_self();
    printf(
        "Tid: %p \033[91mBegin \033[91m>>>>UP>>>>\033[0m upNum %d downNum:
"
        "%d upTotal %d downTotal %d\n",
        tid, upNum, downNum, totalUpNum, totalDownNum);

    for (int i = 0; i < LOCKNUM; ++i) {
        sem_wait(sl[i]);
        printf("Tid: %p \033[91m%d\033[0m Up\n", tid, i + 1);
        sem_post(sl[i]);
        if (!i) sem_post(upBegin);
    }
    --upNum;
    printf("Tid: %p \033[91mship C.I.F. >>>>Uppppp>>>>\033[0m\n", tid);
    if (!upNum) sem_post(mutex);
    return NULL;
}

void *Down(void *a) {
    // up & down sent one ship once
    sem_wait(downBegin);
    if (!downNum) sem_wait(mutex); // flow one direction(no care upNum)
    while (upNum || totalDownNum - totalUpNum > HUNGERNUM) {
        sem_post(mutex);

```

```

        sem_wait(mutex);
    }
    ++totalDownNum;
    ++downNum;
    pthread_t tid = pthread_self();
    printf(
        "Tid: %p \033[93mBegin \033[93mDown\033[0m upNum %d
downNum: %d "
        "upTotal %d downTotal %d\n",
        tid, upNum, downNum, totalUpNum, totalDownNum);

    for (int i = LOCKNUM - 1; i >= 0; --i) {
        sem_wait(sl[i]);
        printf("Tid: %p \033[93m%d\033[0m Down\n", tid, i + 1);
        sem_post(sl[i]);
        if (i == LOCKNUM - 1) sem_post(downBegin);
    }
    --downNum;
    printf("Tid: %p \033[93mship C.I.F. \033[93mDownnnnnnn\033[0m\n",
tid);
    if (!downNum) sem_post(mutex);
    return NULL;
}

void thread_one_create(pthread_t nowTid, int types) {
    if (types)
        pthread_create(&nowTid, NULL, Up, NULL);
    else
        pthread_create(&nowTid, NULL, Down, NULL);
}

void thread_one_join(pthread_t nowTid) { pthread_join(nowTid, NULL); }

int main(int argc, char *argv[]) {
    printf(
        "\033[01;34m Write by Jiang Huiqiang 1801210840 in 2019-04-21
\033[0m\n");
    sem_unlink("sem");
    for (int i = 0; i < LOCKNUM; ++i) sl[i] = sem_open("sem", O_CREAT,
0, 1);
    upBegin = sem_open("sem", O_CREAT, 0, 1);
    downBegin = sem_open("sem", O_CREAT, 0, 1);
    mutex = sem_open("sem", O_CREAT, 0, 1);

    for (int i = 0; i < THREADNUM; ++i) thread_one_create(tidList[i], i
% 2);
    for (int i = 0; i < THREADNUM; ++i) thread_one_join(tidList[i]);

    for (int i = 0; i < LOCKNUM; ++i) sem_destroy(sl[i]);

    pthread_exit(NULL);
}

```



```
2. gunjianpan@Mr-J-MacBook-Pro: ~ (zsh)
Tid: 0x70001226000 ship C.I.F. >>>>Uppppp>>>>
Tid: 0x700012472000 1 Up
Tid: 0x700012472000 2 Up
Tid: 0x700012472000 3 Up
Tid: 0x700012472000 4 Up
Tid: 0x700012472000 5 Up
Tid: 0x700012472000 ship C.I.F. >>>>Uppppp>>>>
Tid: 0x700011ed1000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 2 upTotal 13 downTotal 9
Tid: 0x700011ed1000 5 Down
Tid: 0x7000123ef000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 3 upTotal 13 downTotal 10
Tid: 0x7000123ef000 5 Down
Tid: 0x700011ed1000 4 Down
Tid: 0x7000123ef000 4 Down
Tid: 0x7000123ef000 3 Down
Tid: 0x7000120dd000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 4 upTotal 13 downTotal 11
Tid: 0x700011ed1000 3 Down
Tid: 0x7000123ef000 2 Down
Tid: 0x700011ed1000 2 Down
Tid: 0x7000123ef000 1 Down
Tid: 0x700011dcb000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 1 upTotal 13 downTotal 8
Tid: 0x700011ed1000 1 Down
Tid: 0x700011dcb000 5 Down
Tid: 0x7000120dd000 5 Down
Tid: 0x700011ed1000 ship C.I.F. <<<<Downnnnnnn<<<<
Tid: 0x700011dcb000 4 Down
Tid: 0x700011dcb000 3 Down
Tid: 0x700011dcb000 2 Down
Tid: 0x700011dcb000 1 Down
Tid: 0x700011dcb000 ship C.I.F. <<<<Downnnnnnn<<<<
Tid: 0x7000123ef000 ship C.I.F. <<<<Downnnnnnn<<<<
Tid: 0x7000120dd000 4 Down
Tid: 0x7000120dd000 3 Down
Tid: 0x7000120dd000 2 Down
Tid: 0x7000120dd000 1 Down
Tid: 0x7000120dd000 ship C.I.F. <<<<Downnnnnnn<<<<
Tid: 0x70000fe94000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 1 downNum: 0 upTotal 14 downTotal 11
Tid: 0x70000fe94000 1 Up
Tid: 0x70000fe94000 2 Up
Tid: 0x70000fe94000 3 Up
Tid: 0x70000fe94000 4 Up
Tid: 0x70000fe94000 5 Up
Tid: 0x70000fe94000 ship C.I.F. >>>>Uppppp>>>>
Tid: 0x700011e4e000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 4 downNum: 0 upTotal 17 downTotal 11
Tid: 0x7000fd8e000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 2 downNum: 0 upTotal 15 downTotal 11
Tid: 0x700011c42000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 5 downNum: 0 upTotal 18 downTotal 11
Tid: 0x700011e4e000 1 Up
Tid: 0x700011e4e000 2 Up
Tid: 0x700011e4e000 3 Up
Tid: 0x700011e4e000 4 Up
Tid: 0x700011e4e000 5 Up
```

解决 信号量维护 HUGGER

```

/*
 * @Author: gunjianpan
 * @Date: 2019-04-21 16:36:14
 * @Last Modified by: gunjianpan
 * @Last Modified time: 2019-04-21 22:46:23
 */
#include <pthread.h>
#include <semaphore.h>
#include <stdio.h>
#include <stdlib.h>
#define THREADNUM 100
#define LOCKNUM 5
#define HUNGERNUM 5
int upNum = 0, downNum = 0;
int totalUpNum = 0, totalDownNum = 0;
sem_t *sl[LOCKNUM], *mutex, *upBegin, *downBegin, *hungerUpper,
*hungerDown;
pthread_t tidList[THREADNUM];

void *Up(void *b) {
    // up & down sent one ship once
    sem_wait(upBegin);
    if (!upNum) sem_wait(mutex); // flow one direction(no care downNum)
    while (downNum || totalUpNum - totalDownNum > HUNGERNUM) {
        sem_post(mutex);
        sem_wait(mutex);
    }
    sem_wait(hungerUpper);
    sem_post(hungerDown);
    ++totalUpNum;
    ++upNum;
    pthread_t tid = pthread_self();
    printf(
        "Tid: %p \033[91mBegin \033[91m>>>>UP>>>>\033[0m upNum %d downNum:
"
        "%d upTotal %d downTotal %d\n",
        tid, upNum, downNum, totalUpNum, totalDownNum);

    for (int i = 0; i < LOCKNUM; ++i) {
        sem_wait(sl[i]);
        printf("Tid: %p \033[91m%d\033[0m Up\n", tid, i + 1);
        sem_post(sl[i]);
        if (!i) sem_post(upBegin);
    }
    --upNum;

    printf("Tid: %p \033[91mship C.I.F. >>>>Uppppp>>>>\033[0m\n", tid);
    if (!upNum) sem_post(mutex);
    return NULL;
}

void *Down(void *a) {
    // up & down sent one ship once

```

```

sem_wait(downBegin);
if (!downNum) sem_wait(mutex); // flow one direction(no care upNum)
while (upNum || totalDownNum - totalUpNum > HUNGERNUM) {
    sem_post(mutex);
    sem_wait(mutex);
}
sem_wait(hungerDown);
sem_post(hungerUpper);
++totalDownNum;
++downNum;
pthread_t tid = pthread_self();
printf(
    "Tid: %p \033[93mBegin \033[93mDown\033[0m upNum %d
downNum: %d "
    "upTotal %d downTotal %d\n",
    tid, upNum, downNum, totalUpNum, totalDownNum);

for (int i = LOCKNUM - 1; i >= 0; --i) {
    sem_wait(sl[i]);
    printf("Tid: %p \033[93m%d\033[0m Down\n", tid, i + 1);
    sem_post(sl[i]);
    if (i == LOCKNUM - 1) sem_post(downBegin);
}
--downNum;
printf("Tid: %p \033[93mship C.I.F. <<<<Downnnnnn<<<<\033[0m\n",
tid);
if (!downNum) sem_post(mutex);
return NULL;
}

void thread_one_create(pthread_t nowTid, int types) {
    if (types)
        pthread_create(&nowTid, NULL, Up, NULL);
    else
        pthread_create(&nowTid, NULL, Down, NULL);
}

void thread_one_join(pthread_t nowTid) { pthread_join(nowTid, NULL); }

int main(int argc, char *argv[]) {
    printf(
        "\033[01;34m Write by Jiang Huiqiang 1801210840 in 2019-04-21
\033[0m\n");
    sem_unlink("sem");
    for (int i = 0; i < LOCKNUM; ++i) sl[i] = sem_open("sem", O_CREAT,
0, 1);
    upBegin = sem_open("sem", O_CREAT, 0, 1);
    downBegin = sem_open("sem", O_CREAT, 0, 1);
    mutex = sem_open("sem", O_CREAT, 0, 1);
    hungerDown = sem_open("sem", O_CREAT, 0, HUNGERNUM);
    hungerUpper = sem_open("sem", O_CREAT, 0, HUNGERNUM);

    for (int i = 0; i < THREADNUM; ++i) thread_one_create(tidList[i], i
% 2);
    for (int i = 0; i < THREADNUM; ++i) thread_one_join(tidList[i]);

```

```

for (int i = 0; i < LOCKNUM; ++i) sem_destroy(sl[i]);

pthread_exit(NULL);
}

```

```

2. gunjianpan@Mr-J-MacBook-Pro: ~ (zsh)
Tid: 0x700010f77000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x700011bbf000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 3 upTotal 18 downTotal 15
Tid: 0x700011bbf000 5 Down
Tid: 0x700011bbf000 4 Down
Tid: 0x700011bbf000 3 Down
Tid: 0x700011bbf000 2 Down
Tid: 0x700011bbf000 1 Down
Tid: 0x700011bbf000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x700012807000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 2 upTotal 18 downTotal 16
Tid: 0x700012807000 5 Down
Tid: 0x700012807000 4 Down
Tid: 0x700012807000 3 Down
Tid: 0x700012807000 2 Down
Tid: 0x700012807000 1 Down
Tid: 0x700012807000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x7000117a7000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 2 upTotal 18 downTotal 17
Tid: 0x7000117a7000 5 Down
Tid: 0x7000117a7000 4 Down
Tid: 0x7000117a7000 3 Down
Tid: 0x7000117a7000 2 Down
Tid: 0x7000117a7000 1 Down
Tid: 0x7000117a7000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x70001001d000 Begin ++<<<<Down<<<<+++ upNum 0 downNum: 1 upTotal 18 downTotal 12
Tid: 0x70001001d000 5 Down
Tid: 0x70001001d000 4 Down
Tid: 0x70001001d000 3 Down
Tid: 0x70001001d000 2 Down
Tid: 0x70001001d000 1 Down
Tid: 0x70001001d000 ship C.I.F. <<<<Downnnnnn<<<<
Tid: 0x70001161e000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 6 downNum: 0 upTotal 24 downTotal 17
Tid: 0x700011724000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 1 downNum: 0 upTotal 19 downTotal 17
Tid: 0x70001161e000 1 Up
Tid: 0x7000107ca000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 2 downNum: 0 upTotal 20 downTotal 17
Tid: 0x70001130c000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 5 downNum: 0 upTotal 23 downTotal 17
Tid: 0x700010ce8000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 4 downNum: 0 upTotal 22 downTotal 17
Tid: 0x70001161e000 2 Up
Tid: 0x70001161e000 3 Up
Tid: 0x700011724000 1 Up
Tid: 0x7000102ac000 Begin ΔΔΔ>>>>UP>>>>ΔΔΔ upNum 3 downNum: 0 upTotal 21 downTotal 17
Tid: 0x70001161e000 4 Up
Tid: 0x7000107ca000 1 Up
Tid: 0x7000107ca000 2 Up
Tid: 0x7000107ca000 3 Up
Tid: 0x70001130c000 1 Up
Tid: 0x700011724000 2 Up
Tid: 0x700010ce8000 1 Up
Tid: 0x70001161e000 5 Up
Tid: 0x7000102ac000 1 Up
Tid: 0x70001161e000 ship C.I.F. >>>>Uppppp>>>>
Tid: 0x7000102ac000 2 Up

```