Report

—Project 2 : Thread-Safe Malloc

Student Name: Yitong Wang
Student NetID: yw471

---

1. Overview of implementation

In this homework, what I have done is based on the homework1 bf_malloc and free code. By reading the materials on the internet, I find there are many ways to implement it and finally I choose a more simple and efficient way: only keep free node in the linked list, which meanest is a free linked list.
No matter which version, we should initialize a lock first by using following code:

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER;
```

1.1 Lock Version

For lock version, we can use lock before and after the malloc and free, which make the linked list can only be accessed by one thread at once. The code is as below:

```
void *ts_malloc_lock(size_t size){
    pthread_mutex_lock(&lock);
    int lock_version = 0;
    void * ptr = bf_malloc(size, lock_version, &head_lock, &tail_lock);
    pthread_mutex_unlock(&lock);
    return ptr;
}
```

The code between lock and unlock is the critical section.

1.2 Non-Lock Version

For this version, we can use lock before and after the sbrk() function, which achieved sbrk() concurrency. I use Thread Local Storage. The code is as below:

```
void *ts_malloc_nolock(size_t size){
    int lock_version = 1;
    void * ptr = bf_malloc(size,lock_version, &head_nolock, &tail_nolock);
```

```
    return ptr;
}
void ts_free_nolock(void *ptr){
    my_free(ptr,&head_nolock,&tail_nolock);
}
```

---

2. Result & Analysis

2.1 Test Results

VERSION	Average Execution Time	Average Data Segment Size	Pass Rate
LOCK	0.25 s	42342368 bytes	20/20
UNLOCK	0.225 s	43041000 bytes	20/20

The test result are above, and all test were passed. For the execution time and data segment size, every time I run the code it is different, but the time have significantly difference while the data segment size is nearly the same. In theory, the non lock version's execution time will short than the lock version because it only lock the sbrk() this function When doing this homework I do meet some problems and meeting some problem which is left by homework1.