4108056051 資工三 鄭穎

1. 標頭檔、全域變數、函式名稱

```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#include <time.h>
sem_t max_buffer_full;
sem_t min_buffer_full;
sem_t max_buffer_mutex;
sem_t min_buffer_mutex;
int big_buffer[1024];
int max_buffer[4],min_buffer[4];
int maximum=0;
int minimum=2147483647;
int index_maxP=-1;
int index_minP=-1;
void *producer(void*);
void *max_consumer();
void *min_consumer();
int main(){
```

2. Main function

```c
int main(){
        time_t t;
        srand((unsigned)time(&t));
        for(int i=0;i<1024;i++)                 //generate random number
                big_buffer[i]=rand();
        sem_init(&max_buffer_full,0,0);         //initialize semaphore
        sem_init(&min_buffer_full,0,0);
        sem_init(&max_buffer_mutex,0,1);
        sem_init(&min_buffer_mutex,0,1);

        pthread_t p[4];
        pthread_t c1,c2;

        for(int i=0;i<4;i++){                   //create producer threads
                int *arg=malloc(sizeof(*arg));
                *arg=(i+1);
                pthread_create(&p[i],NULL,producer,arg);
        }
        pthread_create(&c1,NULL,max_consumer,NULL);//create consumer threads
        pthread_create(&c2,NULL,min_consumer,NULL);

        for(int i=0;i<4;i++)
                pthread_join(p[i],NULL);
        pthread_join(c1,NULL);
        pthread_join(c2,NULL);

        sem_destroy(&max_buffer_full);          //destroy threads
```

```c
        sem_destroy(&min_buffer_full);
        sem_destroy(&max_buffer_mutex);
        sem_destroy(&min_buffer_mutex);

        printf("Success! maximum=%d and minimum=%d\n",maximum,minimum);
        return 0;
}
```

3. producer function

```
void *producer(void *input){
        int num=*((int *)input);        //distinguish producers
        int l,u;
        int tmp_max=0;
        int tmp_min=2147483647;
        l=256*(num-1);                  //decide access range
        u=l+256;
        for(int i=l;i<u;i++){           //find tmp max and min
                if(tmp_max<big_buffer[i])
                        tmp_max=big_buffer[i];
                if(tmp_min>big_buffer[i])
                        tmp_min=big_buffer[i];
        }
        printf("Temporary max=%d and min=%d\n",tmp_max,tmp_min);
        sem_wait(&max_buffer_mutex);    //store tmp max to max_buffer
        index_maxP++;
        max_buffer[index_maxP]=tmp_max;
        printf("Producer: Put %d into max_buffer at %d\n",tmp_max,index_maxP);
        sem_post(&max_buffer_mutex);
        sem_post(&max_buffer_full);

        sem_wait(&min_buffer_mutex);    //store tmp min to min_buffer
        index_minP++;
        min_buffer[index_minP]=tmp_min;
        printf("Producer: Put %d into min_buffer at %d\n",tmp_min,index_minP);
        sem_post(&min_buffer_mutex);
```

```
        sem_post(&min_buffer_full);

        pthread_exit(NULL);
}
```

4. max_consumer function

```
void *max_consumer(){
        for(int i=0;i<4;i++){           //iterate max_buffer
                sem_wait(&max_buffer_full);     //use semaphore to protect
                if(maximum<max_buffer[i])
                        maximum=max_buffer[i];  //update maximum
                printf("Update! maximum=%d\n",maximum);
        }
        pthread_exit(NULL);
}
```

5. min_consumer function

```
void *min_consumer(){
        for(int i=0;i<4;i++){           //iterate min_buffer
                sem_wait(&min_buffer_full);     //use semaphore to protect
                if(minimum>min_buffer[i])
                        minimum=min_buffer[i];  //update minimum
                printf("Update! minimum=%d\n",minimum);
        }
        pthread_exit(NULL);
}
```

6. 執行結果

```
cs4108056051@cs4108056051-VirtualBox:~$ ./hw2
Temporary max=2147469841 and min=6939507
Producer: Put 2147469841 into max_buffer at 0
Producer: Put 6939507 into min_buffer at 0
Update! maximum=2147469841
Temporary max=2118421993 and min=2416949
Producer: Put 2118421993 into max_buffer at 1
Update! maximum=2147469841
Temporary max=2143124030 and min=19485054
Producer: Put 2143124030 into max_buffer at 2
Temporary max=2135019593 and min=6072641
Producer: Put 2135019593 into max_buffer at 3
Update! minimum=6939507
Update! maximum=2147469841
Update! maximum=2147469841
Producer: Put 2416949 into min_buffer at 1
Producer: Put 19485054 into min_buffer at 2
Update! minimum=2416949
Update! minimum=2416949
Producer: Put 6072641 into min_buffer at 3
Update! minimum=2416949
Success! maximum=2147469841 and minimum=2416949
```