4. (40 points) Implement program *accumulator* and *sendnum*. The program accumulates the numbers (long integer) from multiple Message Queues, then print it to standard output. The program uses one thread per message queue. The number of message queues is set by command-line option -q. The default value is 1.

```
$ ./accumulator -q 2 &
[1] 26721
queue ID 0 is 98307
queue ID 1 is 131076
$ ./sendnum 0x100 50
sum is 50.
$ ./sendnum 0x101 30
sum is 80.
```

```
#include "opue.h"   //sendnum.
                    发送消息
struct msg {
    long mytype;
    long value;
};
int main(int argc, char *argv[]){
    key_t key;
    long value;
    int id;
    struct msg m;
if(argc != 3){ err_sys("usage: %s key value\n",a
            argv[0]);
    sscanf(argv[1], "%x", &key);
    sscanf(argv[2], "%ld", &value);
if((id = msgget(key, 0)) < 0){ err_sys("error: ma...  fail to get");}

m.mtype = 1;
m.value = value;
  if(msgsnd(id, &m, sizeof(long), 0) < 0){
    err_sys("Fail to send msg"); }

  return 0;
}.
```