```
#include <pthread.h>

typedef struct job
{
    int jid;          /* job ID */
    pthread_t tid;    /* thread ID */
    char *cmd;        /* job command */
    char *stat;       /* job status */
    int estat;        /* job exit status */
    char *start;      /* job start time */
    char *stop;       /* job stop time */
    char fnout[10];   /* filename where job stdout is redirected */
    char fnerr[10];   /* filename where job stderr is redirected */
} job;

typedef struct queue
{
    int size;         /* maximum size of the queue */
    job **buffer;     /* queue buffer */
    int start;        /* index to the start of the queue */
    int end;          /* index to the end of the queue */
    int count;        /* number of elements in the queue */
} queue;

job create_job(char *cmd, int jid);
void list_jobs(job *jobs, int n, char *mode);

queue *queue_init(int n);
int queue_insert(queue *q, job *jp);
job *queue_delete(queue *q);
void queue_destroy(queue *q);

int get_line(char *s, int n);
int is_space(char c);
char *left_strip(char *s);
char *get_copy(char *s);
char *get_copy_until_newline(char *s);
char *current_datetime_str();
char **get_args(char *line);
int open_log(char *fn);
```