From rb.h

```
typedef struct rb_node {
  ...
  ...
  union {
    int ikey;
    char *key;
  } k;
  union {
    char *val;
  } v;
  ...
} *Rb_node;
```

- `Rb_node make_rb()`: Mallocs and returns a new rb-tree.
- `void rb_inserti(Rb_node t, char *key, char *val)`: Creates and inserts a new rb-node into the tree `t` with key `key` and val `val`. The key is an integer, and nodes are sorted by integer value.

From dlist.h

```
typedef struct dlist {
  struct dlist *flink;
  struct dlist *blink;
  int val;
} *Dlist;
```

- `Dlist make_dl()`: Mallocs and returns a new dlist.
- `void dl_insert_b(Dlist node, int val)`: Makes a new node, and inserts it into the list before the node `node`. If `node` is the head of the list, the new node is inserted at the end of the list.
- `void dl_delete_node(Dlist node)`: Deletes node `node` from whatever `Dlist` it is in.

System Call Syntax

- `int fork()`
- `int dup2(int fd1, int fd2)`
- `int pipe(int fds[2])`
- `int close(int fd)`