Linked lists

list.h

```
1  struct list {
2          struct list *next;
3          char *value;
4  };

5  struct list *list_add(struct list *list, char *value);
6  void list_dump(struct list *list);
7  void list_free(struct list *list);
```

list.c

```
1  #include <stdlib.h>
2  #include "list.h"

3  int main(int argc, char **argv)
4  {
5          struct list *list = NULL;

6          list = list_add(list, "hello");
7          list = list_add(list, "hello");
8          list = list_add(list, "world");
9          list = list_add(list, "lion");

10         list_dump(list);
11         list_free(list);

12         return 0;
13 }
```

list_add.c

```c
#include <stdlib.h>
#include <string.h>
#include "list.h"

struct list *list_add(struct list *list, char *value)
{
        struct list *tmp;

        if (!list || strcmp(list->value, value) > 0) {
                tmp = malloc(sizeof *tmp);
                tmp->next = list;
                tmp->value = value;
                return tmp;
        }

        if (strcmp(list->value, value) < 0)
                list->next = list_add(list->next, value);

        return list;
}
```

list_dump.c

```c
#include <stdio.h>
#include "list.h"

void list_dump(struct list *list)
{
        while (list) {
                printf("%s\n", list->value);
                list = list->next;
        }
}
```

list_free.c

```c
#include <stdlib.h>
#include "list.h"

void list_free(struct list *list)
{
        if (list->next)
                list_free(list->next);
        free(list);
}
```

```
$ ./list
hello
lion
world
```