

List.h

```
struct Node {  
    char *name;  
    void *attribute;  
    struct Node *next;  
};
```

```
struct List {  
    struct Node* head;  
};
```

```
struct List * CreateList();
```

```
void insert(struct List *aList, char *name, struct Node ** element);
```

```
struct Node * find(struct List *aList, char *name);
```

```
void DestroyList(struct List * aList);
```

List.c

```
#include "List.h"  
#include <string.h>
```

```
struct List * CreateList() {  
    struct List * aList;  
    aList = (struct List *) malloc(sizeof(struct List));  
    aList->head = NULL;  
    return aList;  
}
```

```
void insert(struct List *aList, char *name, struct Node ** element) {  
    struct Node *n;  
    n = (struct Node *) malloc(sizeof(struct Node));  
    n->name = (char *) malloc(strlen(name)+1);  
    strcpy(n->name, name);  
    n->next = aList->head;  
    aList->head = n;  
    *element = n;  
}
```

List.c

```
struct Node * find(struct List *aList, char *name) {
    struct Node *temp;
    temp = aList->head;
    while (temp != NULL) {
        if (strcmp(temp->name, name) == 0)
            return temp;
        temp = temp->next;
    }
    return NULL;
}
```

List.c

```
void DestroyList(struct List * aList) {
    struct Node * temp;
    struct Node * temp1;
    temp = aList->head;
    while (temp != NULL) {
        temp1 = temp->next;
        free(temp->name);
        free(temp);
        temp = temp1;
    }
    free(aList);
}
```

main.c

```
#include <stdio.h>
#include "List.h"

int main (int argc, const char * argv[]) {
    struct List *x;
    struct Node *w;
    char name[50];
    FILE *f;
    x = CreateList();
    if ((f = fopen("input1", "r")) == NULL) exit(3);
    while (fgets(name, 50, f) != NULL) {
        name[strlen(name)-1] = '\0'; //remove the newline
        insert(x, name, &w);
    }
}
```

main.c

```
if (find(x, "tom") != NULL) printf("tom found\n");  
if (find(x, "jake") != NULL) printf("jake found\n");
```

```
if (find(x, "martin") != NULL) printf("martin found\n");  
if (find(x, "sam") != NULL) printf("sam found\n");
```

```
    DestroyList(x);  
return 0;  
}
```