VE482 Lab Report

Lab 5 - Fall 2020

Name: Wu Qinhang
ID: 518370910041
Email: william_wu@sjtu.edu.cn

Table of Contents

- Layer Programming
- Libraries

Layer Programming
File Structure
I/O File Format for cmd ui
Compile the Program
Libraries
static library
dynamic library

Layer Programming

The program can be divided into three layers, what are they?

- list structure
- sorting operations
- user interface

File Structure

- `list.c/h`
  - list structure
- `sort.c/h`
  - sorting operations
- `ui.c/h`
  - user interface
- `driver_cmd.c`
  - main function for cmd user interface
- `driver_menu.c`
  - main function for menu user interface

I/O File Format for cmd ui

Input:

- `rand_int.txt` : the type of data is `int`
- `rand_char.txt` : the type of data is `char*`
- `rand_double.txt` : the type of data is `double`

Output:

Format: `sortingtype_datatype.txt`

sortingtype:

- `rand` : the order of elements is random
- `inc` : the sorting order is increasing
- `dec` : the sorting order is decreasing

datatype:

- `int` : the type of data is `int`
- `char` : the type of data is `char*`
- `double` : the type of data is `double`

Compile the Program

A `Makefile` is provided to compile the program. A sample command is provided below:

```
make cmd
./ui_cmd rand_int.txt dec
make menu
./ui_menu
```

A set of testcases is provided in the folder.

Libraries

- **What are the three stages performed when compiling a file?**
  - preprocessing, compilation and linking
  - For preprocessing, the preprocessor includes other header files, expands macros, does conditional compilation and so on.
  - For compilation, the compiler read the source files and produce the corresponding assembly code.
  - For linking, the linker takes several object files and produce a single binary executable program.
- **difference between static and dynamic library**
  - static libraries are locked in to the program when being compiled; dynamic libraries exist as separate files outside the binary file.

static library

use `-c` command to compile the source and header file so as to create a static library file.

create two static library files:

```
gcc -c list.c -o list.o
ar rcs list.a list.o
gcc -c sort.c -o sort.o
ar rcs sort.a sort.o
gcc -c ui.c -o ui.o
ar rcs ui.a ui.o
```

compile the cmd driver program:

```
gcc -c driver_menu.c -o ui_cmd.o
gcc -o ui_cmd ui_cmd.o -L. -lsort -llist -lui
./ui_cmd
```

dynamic library [1]

```
gcc *.c -c -fPIC
gcc *.o -shared -o libmenu.so
gcc -L. -o ui_menu ui_menu.c -lmenu
export LD_LIBRARY_PATH=./:$LD_LIBRARY_PATH
./ui_menu
```

difference between a library and the API:

An API is a software component that provides the developer with many applicable utilities, while a library is a collection of functionality that may not make up of an application.

For the implementation, see `./lab5_dlist.h`.

---

1. "Shared libraries with GCC on Linux - Cprogramming.com," *Cprogramming.com* , 2019. https://www.cprogramming.com/tutorial/shared-libraries-linux-gcc.html (accessed Oct. 28, 2020). ↩