

C++夏令营进度汇报

李瑞堃

总体进度

- ▶ 主程序：git.cpp
- ▶ 实现功能：cmdline、diff、commit、add、rm、mv

```
self-git > h git.h > ...
1  #include <stdlib.h>
2  #include <stdio.h>
3  #include <string>
4  #include <string.h>
5  #include <unistd.h>
6  #include <dirent.h>
7  #include <sys/stat.h>
8  #include <sys/types.h>
9
10 #define MAX_ACTION_NUM 30 //最大动作数量
11 #define MAX_SHORT_OPT_NUM 5 //每个动作最大短选项数量
12 #define MAX_LONG_OPT_NUM 5 //每个动作最大长选项数量
13 #define MAX_PARAM_NUM 5 //每个选项最大参数数量
14
15 #define DIFF_MAX_ROW_NUM 200 //diff读取的文件的最大行数s
16
17 bool file_copy(char old_file[], char new_file[]); //文件copy函数的声明
```

Cmdline

类定义:

```
class cmdl
{
public:
    char action_name[MAX_ACTION_NUM][301];           //用于存储动作名称, 目前最多30个
    char action_help[MAX_ACTION_NUM][301];           //用于存储动作的说明
    char action_short_options[MAX_ACTION_NUM][MAX_SHORT_OPT_NUM]; //用于存储每个动作的短选项, 目前最多5个
    char action_long_options[MAX_ACTION_NUM][MAX_LONG_OPT_NUM][301]; //用于存储每个动作的长选项, 目前最多5个
    int action_option_num[MAX_ACTION_NUM][2];         //用于存储每个动作的长/短选项的数量, [0]存短选项、[1]存长选项
    int action_num;                                   //已存储动作的数量

    std::string action_now;                           //从命令行获取的本轮动作
    char short_option_now;                             //从命令行获取的动作短选项
    char long_option_now[301];                         //从命令行获取的动作长选项
    char action_help_now[301];                         //当前动作的说明
    char action_param_now[MAX_PARAM_NUM][301];         //当前动作的选项的参数, 目前最多5个
    int action_param_num;                             //当前动作的选项的参数数量

    cmdl(); // 类的构造函数
    ~cmdl(); // 类的析构函数

    bool add_action(const char *action, const char short_opt, const char *long_opt, const int param_num, const char *help);
    bool parse(const int argc, char *argv[]);
};
```

Cmdline

效果演示：

```
cmdl cmd;  
cmd.add_action("diff", ' ', "", 2, "Show changes between param_file1 and param_file2");  
cmd.add_action("add", ' ', "", 2, "submit the files from work dir to stage dir");  
cmd.add_action("commit", 'm', "message", 1, "commit the files from stage dir to local dir with the parameter as message");  
cmd.parse(argc, argv);
```

```
lrk@lrk-virtual-machine:~/桌面/C++/self-git$ ./git diff  
命令行解析:  
action: diff  
help: Show changes between param_file1 and param_file2  
no short option  
no long option  
no parameter  
命令行解析结束
```

```
lrk@lrk-virtual-machine:~/桌面/C++/self-git$ ./git commit -m just-for-test  
命令行解析:  
action: commit  
help: commit the files from stage dir to local dir with the parameter as message  
short option: m  
no long option  
1 parameter(s): just-for-test  
命令行解析结束
```

Diff

预期效果：输入指定两文件，比较增删；后期补上 work_dir 与 stage 的差异对比功能

思路：1. 逐行比较
2. LCS算法

效果演示：

```
Lrk@Lrk-virtual-machine:~/桌面/C++/self-git$ ./git diff READMEa.md READMEb.md
```

```
命令行解析：
```

```
action: diff
```

```
help: Show changes between param_file1 and param_file2
```

```
no short option
```

```
no long option
```

```
2 parameter(s): READMEa.md      READMEb.md
```

```
命令行解析结束
```

```
git diff READMEa.md READMEb.md
```

```
# cpp_summer_camp_code
```

```
#### 介绍
```

```
Dian summer camp
```

```
#### 日志
```

```
##### 2021-7-23
```

```
增加了命令行解析功能，可以获取命令行输入的action、option、parameter
```

```
目前存在的问题：
```

```
1. 创立新的action时未对其选项的参数个数进行限制
```

```
2. 命令行输入不同类型指令时，有些指令会出bug，有待调整
```

```
##### 2021-7-24
```

```
修复了昨天的问题 2
```

```
##### 2021-7-25
```

```
1. 增加了 diff 功能，因为今天有别的任务所以采取的是逐行对比简单思想，没有使用求最长公共子序列LCS算法思想来做
```

```
1. 增加了 diff 功能，因为今天有别的任务所以采取的是逐行对比想，没有使用求最长公共子序列LCS算法思想来做
```

```
2. 具体函数定义和使用的库、宏都以静态库形式链接到主程序 git.cpp 中
```

```
2. 具体函数定义和使用的库、宏都以静态库形式链接到哈哈哈哈哈主程序 git.cpp 中
```

```
3. 再次修复了cmdline中的某些bug，并解决了23号的问题1
```


Add

预期效果：从工作目录向暂存区拷贝文件

效果演示：

```
lrk@lrk-virtual-machine:~/桌面/C++/self-git$ ./git add /home/lrk/桌面/C++/self-git/work_dir /home/lrk/桌面/C++/self-git/stage_dir
```

命令行解析：

action: add

help: submit the files from work dir to stage dir

no short option

no long option

2 parameter(s): /home/lrk/桌面/C++/self-git/work_dir /home/lrk/桌面/C++/self-git/stage_dir

命令行解析结束

```
git add /home/lrk/桌面/C++/self-git/work_dir /home/lrk/桌面/C++/self-git/stage_dir
```

工作目录： /home/lrk/桌面/C++/self-git/work_dir

暂存目录： /home/lrk/桌面/C++/self-git/stage_dir

/home/lrk/桌面/C++/self-git/work_dir/Cpp_package_client.cpp	--->	/home/lrk/桌面/C++/self-git/stage_dir/Cpp_package_client.cpp
/home/lrk/桌面/C++/self-git/work_dir/Cpp_package_server.cpp	--->	/home/lrk/桌面/C++/self-git/stage_dir/Cpp_package_server.cpp
/home/lrk/桌面/C++/self-git/work_dir/generate_makefile.c	--->	/home/lrk/桌面/C++/self-git/stage_dir/generate_makefile.c
/home/lrk/桌面/C++/self-git/work_dir/Server.cpp	--->	/home/lrk/桌面/C++/self-git/stage_dir/Server.cpp
/home/lrk/桌面/C++/self-git/work_dir/Client.cpp	--->	/home/lrk/桌面/C++/self-git/stage_dir/Client.cpp

```
git add finish
```

Commit

预期效果：从暂存区向本地仓库移动文件

效果演示：

```
lrk@lrk-virtual-machine:~/桌面/C++/self-git$ ./git commit --message for-test /home/lrk/桌面/C++/self-git/stage_dir /home/lrk/桌面/C++/self-git/local_dir
```

命令行解析：

action: commit

help: commit the files from stage dir to local dir with the parameter as message

no short option

long option: message

3 parameter(s): for-test /home/lrk/桌面/C++/self-git/stage_dir /home/lrk/桌面/C++/self-git/local_dir

命令行解析结束

```
git commit --message for-test /home/lrk/桌面/C++/self-git/stage_dir /home/lrk/桌面/C++/self-git/local_dir
```

message: for-test

暂存目录: /home/lrk/桌面/C++/self-git/stage_dir

本地仓库: /home/lrk/桌面/C++/self-git/local_dir

```
/home/lrk/桌面/C++/self-git/stage_dir/Cpp_package_client.cpp  --->  /home/lrk/桌面/C++/self-git/local_dir/Cpp_package_client.cpp
/home/lrk/桌面/C++/self-git/stage_dir/Cpp_package_server.cpp  --->  /home/lrk/桌面/C++/self-git/local_dir/Cpp_package_server.cpp
/home/lrk/桌面/C++/self-git/stage_dir/generate_makefile.c      --->  /home/lrk/桌面/C++/self-git/local_dir/generate_makefile.c
/home/lrk/桌面/C++/self-git/stage_dir/Server.cpp               --->  /home/lrk/桌面/C++/self-git/local_dir/Server.cpp
/home/lrk/桌面/C++/self-git/stage_dir/Client.cpp               --->  /home/lrk/桌面/C++/self-git/local_dir/Client.cpp
```

```
git commit finish
```

Rm

预期效果：删除暂工作目录和暂存区的指定文件

效果演示：

```
lrk@lrk-virtual-machine:~/桌面/C++/self-git$ ./git rm /home/lrk/桌面/C++/self-git/work_dir/Server.cpp
```

命令行解析：

action: rm

help: remove the file in the work_dir and stage

no short option

no long option

1 parameter(s): /home/lrk/桌面/C++/self-git/work_dir/Server.cpp

命令行解析结束

```
git rm /home/lrk/桌面/C++/self-git/work_dir/Server.cpp
```

```
Work Dir: delete /home/lrk/桌面/C++/self-git/work_dir/Server.cpp succeed!
```

```
warning: /home/lrk/桌面/C++/self-git/work_dir/Server.cpp does not exist in the stage_dir!
```


Mv

预期效果：移动或重命名暂工作目录和暂存区的指定文件

效果演示：

```
git mv _client.cpp Client.cpp
lrk@lrk-virtual-machine:~/桌面/C++/self-git$ ./git mv _client.cpp Client.cpp
命令行解析:
action: mv
help: move or rename a file
no short option
no long option
2 parameter(s): _client.cpp      Client.cpp
命令行解析结束

git mv _client.cpp Client.cpp

_client.cpp      --->      Client.cpp

git mv finished
```