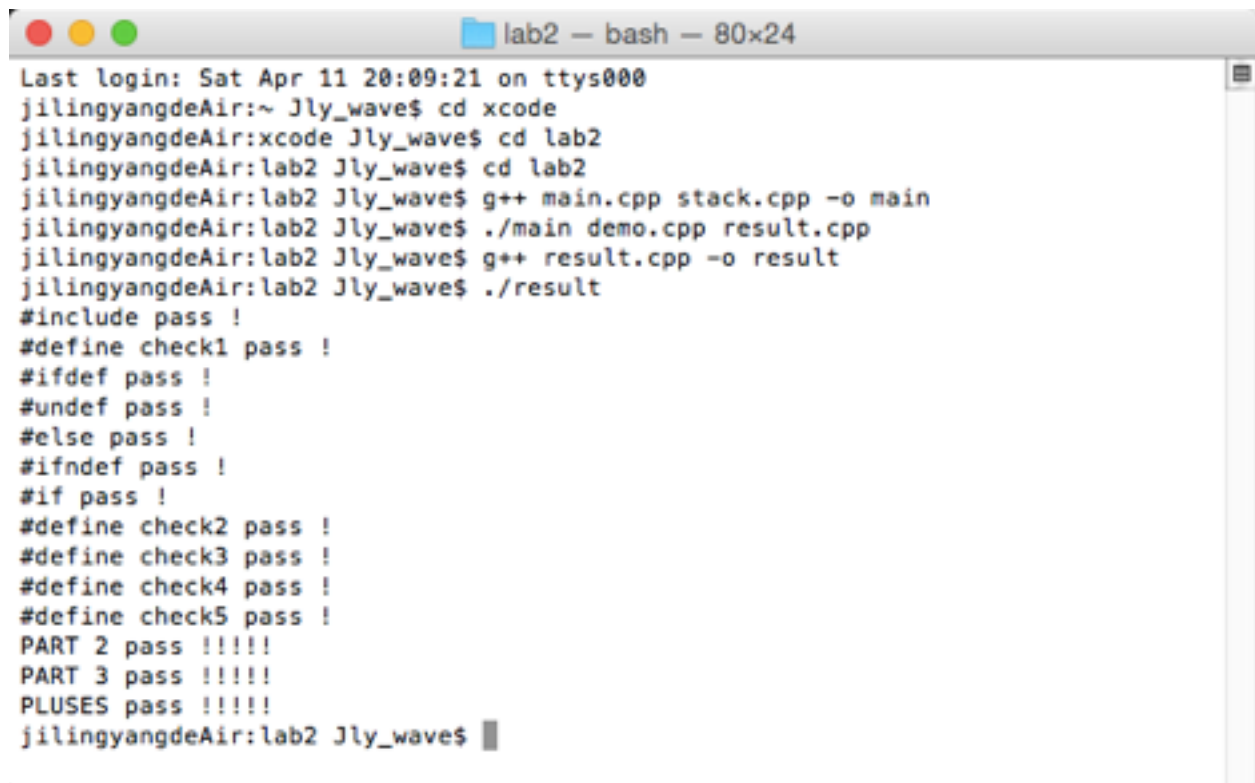


lab 2 说明文档

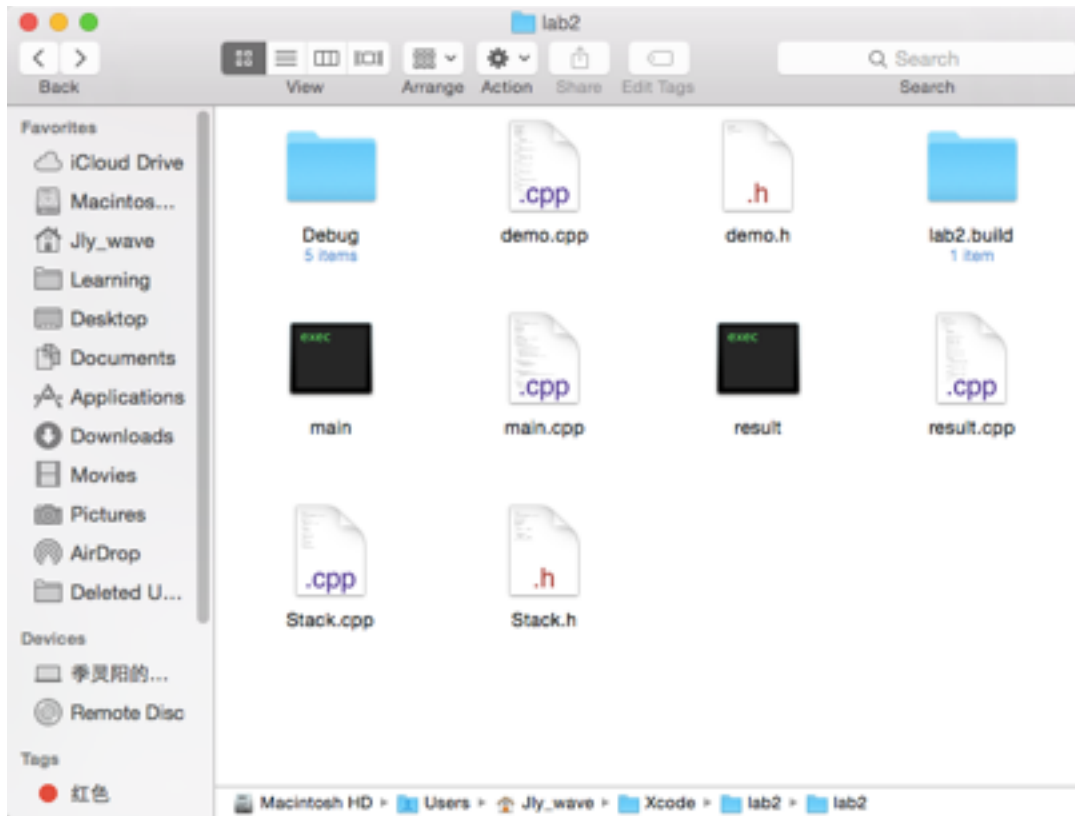
1.文件说明

| | |
|-------------------|---------|
| main.cpp | 主文件 |
| stack.cpp | 栈 |
| stack.h | 栈头文件 |
| lab2 说明文档 | 说明文档 |
| result.cpp | 生成的示例文件 |
| demo.cpp & demo.h | lab测试文件 |

2.用生成文件 result.cpp 进行编译运行后的截图(全过程)



```
lab2 - bash - 80x24
Last login: Sat Apr 11 20:09:21 on ttys000
jilingyangdeAir:~ Jly_wave$ cd xcode
jilingyangdeAir:xcode Jly_wave$ cd lab2
jilingyangdeAir:lab2 Jly_wave$ cd lab2
jilingyangdeAir:lab2 Jly_wave$ g++ main.cpp stack.cpp -o main
jilingyangdeAir:lab2 Jly_wave$ ./main demo.cpp result.cpp
jilingyangdeAir:lab2 Jly_wave$ g++ result.cpp -o result
jilingyangdeAir:lab2 Jly_wave$ ./result
#include pass !
#define check1 pass !
#ifdef pass !
#undef pass !
#else pass !
#endif pass !
#if pass !
#define check2 pass !
#define check3 pass !
#define check4 pass !
#define check5 pass !
PART 2 pass !!!!!
PART 3 pass !!!!!
PLUSES pass !!!!!
jilingyangdeAir:lab2 Jly_wave$
```



3.预编译命令

`#include` 读入指定文件内容

`#include <stack.h>`

`#include "stack.h"`

如果为`<>`表明这个文件为一个工程或标准头文件，查找过程会先检查预定义目录，如果是`"`则是用户提供的头文件。

`#define` 定义宏

`#define abc` 纯定义

`#define a 10` 把a 替换为10

`#define sum(a,b) ((a)+(b))` 把sum(a,b)替换为后者 并且a, b可以是其他有效值

#undef

#undef A 将**#define** 对于A 的定义解绑

#ifdef

#ifdef A 表示若A被定义则。。

#ifndef

#ifndef A 表示若A未被定义则。。

#if

指令检测跟在制造另关键字后的常量表达式。如果表达式为真，则编译后面的代码，知道出现**#else**、**#elif**或**#endif**为止；否则就不编译。

#else

用于中转 **#if** **#ifdef** **#ifndef**

#endif

用于终止**#if** **#ifdef** **#ifndef**

#elif

#elif预处理指令综合了**#else**和**#if**指令的作用。

```
#define DEBUG
```

```
    main()
```

```
    {
```

```
        #ifdef DEBUG
```

```
            cout<<"yes\n"<<endl;
```

```
        #else {
```

```
        }
```

```
        #endif
```

```
        #ifndef DEBUG
```

```
            cout<<"no\n"<<endl;
```

```
        #endif
```

```
    }
```

4.设计思路

*首先我是根据空格来分割各个字符串的

#include .h

直接进行文件读取并且直接放入原文件中

#define

存储一张表（用两个字符串指针）记录define

一个存储修改前，一个存储修改后

对于纯粹定义的 后一个数组存储“true_”或者“false_”代表是否定义

#if

查询define表，进行多次查找直到不再变化，然后进行判断是否为真

#ifdef

直接查表，看是否存在表中且不为“false_”

#ifndef

直接查表，看是否不存在表中或者存在且为“false_”

#else #endif

通过设计一个栈来实现

栈元素存在三种值 0/1/-1

1 表示当前状态为真(当前行可以输出)

0/-1 表示当前状态为假（当前行均不能输出）

*0/1 遇到#else 会进行转换，而-1不会

*遇到if（#if #ifdef #ifndef）根据真假判断0/1 并压入，但如果当前环境已经为0 则直接往栈压入-1

*遇到#endif 则弹出栈顶值

part3（#define 定义一个参数的宏）

首先搜索括号，然后将要处理字符串的括号内元素换成表中括号元素，然后进行普通替换，最后再将原来的()内字符串替换回来。

pluse（实现对#运算符支持）

首先去掉#，然后搜索（）字符，并将其左右加上”替代回去