Topic 2 : 模块化编程

CLion的使用： 标准：c90. c99. c11.

① 新建项目 New Project

IDE左侧：工程结构：

code1：
- main.c
- CMakeLists.txt：构建脚本.
- cmake-build-debug：

构建脚本发展：raw → make → cmake.

② 编程：求圆面积和周长.

在main.c上右键 → New → C/C++ Header File → circle.h
(一般来说.h文件不加入target)

circle.h：double area(double r);

在main.c中：#include "circle.h"，即可以调用 area().

New → C/C++ Source File → 类型选择.c → 加入构建目标：circle.c

```
circle.c：  double area(double r){
                return 3.14*r;
            }
```

③ 这个工程如何工作？

cmake – build – debug 中新增3可执行文件 code1.exe.

code1.dir：临时文件.

Topic 3：构建新的数据类型.

基本数据类型不够用 → 构建新的数据类型.

数据类型 { 数据结构
操作.

构建复数：$x+yi$

① New → C/C++ Header File : complex.h

complex.h：
```
// part 1: data structure
struct complex_t{
    double x;
    double y;
};
// part 2: operations
struct complex_t * complex_add ( struct complex_t *c1,
                                 struct complex_t *c2);
sturuct complex_t *creat (double x, double y);
void print ( struct complex_t *c);
```

② 在 main.c 中加入 #include "complex.h"

③ New → Source File → complex.c

※ C语言 sizeof { sizeof (类型)
sizeof (变量)
sizeof (表达式)

Q1：内存分配之后没有释放会造成什么影响？

Q2：C语言中怎么做内存管理？

Q3：代码中为什么要用指针类型？
如果不用指针类型会怎样？
1. 运行性能：用指针比不用指针高。
指针可以放到物理寄存器里。

2. 支持抽象编程而不是具体编程。

{ CDT：concrete data type
ADT：abstract data type

Q4：为什么要使用 ADT？
※ 数据结构定义不要放在接口，放在实现部分。
放到实现文件之后，为什么.h文件~~还可以实现~~不会报错？

Q5：ADT 可以带来表式独立性

课后练习：实现一个单链表（数据结构、操作）

用本次课介绍的编程思想。