选做题：往年期中考试题（每年助教都会换，助教出题，没有参考价值）

准备：大量看程序、写程序，课后习题做

temp=(a>b)?a:b;输出大的数

字符大小比较：ASCII值

switch：多项分支（锦上添花的，if其实都行）

switch(表达式)

{

case 常量表达式1（与表达式相等继续）:语句序列1;break（断路，到}后面继续运行）;

…

…

case 常量表达式n;语句序列n;break;

default（都不相等）:语句序列n+1;

}

switch表达式：整型、字符型、枚举类型

case互不相同但是顺序无所谓

case后常量表达式只能常量

不加break就匹配后顺序执行不比对直到break，加break就是匹配的执行直接跳出

break：从switch、for、while、do-while语句中跳出终止并跳到后面

单独使用break没有意义

换行提高清晰度（控制显示格式）

构建temp，switch(temp){case0…case2}

多个段的分段函数，switch会清晰

不加break可以合并情况

case 1: case 2: case 3: …; break;：123都干

int的除法向下取整

最关键的是switch表达式的构成问题

复合语句：定义的变量只能在里面用

数值分解：整除的效果

条件运算符

循环结构

一段程序被反复执行有限/无穷次（死循环）

计算机是伟大的傻瓜（按照代码反复执行，速度很快，很忠诚）

检查所有方案：枚举法

当型循环：条件非0→循环体→再判断……直到条件不满足跳出（NS图：左上包围）

直到型循环：循环体→条件非0→再执行……直到条件不满足跳出（NS图：左下包围）

用流程图画也行，但是太松散

三种：while（当型）；do-while（像直到型）；for（综合语句，锦上添花，C之后才有）

循环条件：计数/事件控制

循环体：一条以上用花括号括起来构成复合语句

一定要有改变循环条件的语句，能够终止，不然死循环（计数器加一或减一！）

while:

while(表达式) 循环体语句

表达式非0运行循环体，0跳出

重点：循环条件，循环体构造，计数器和累加器

小心溢出！

连续输入直到回车符，统计字符个数

事件控制循环：

len=0;

(ch=getchar())!=’\n’

getchar()：字符输入（stdio.h，standard input and output），一次只能读一个字符

getch头文件是conio.h（不是编译器写的，非标准C函数，consor input and output）

ENTER=\n\r顺序和操作系统有关

回车→getchar返回10（\n），getch返回13（\r），getchar有回显，get没有

输入到缓冲区buffer，计算机再取数（回车也会放到里面）

清空缓冲区的过程

puts：打印普通字符

函数靠自己记

会的函数太少就会写的很罗嗦（程序员有函数库的书）

缓冲区：键盘录入速度和CPU读取速度的协调

fabs：求绝对值（math.h）

整型除以整型还是整型！

dowhile：

先执行循环语句体一次，再判断表达式

do{循环体}while();

for(<表达式1（初始化）>;<表达式2（条件）>;<表达式3（修改条件）>){循环体}

=表达式1;while（表达式2）{循环体; 表达式3;}

不会忘记初始化、忘记修改条件（如i++），提高安全性，更紧凑，更可读

嵌套循环：内循环必须完全嵌套于外循环中，控制变量不能重名

期中考试无所谓优化，且不允许过度优化

continue短路：提前结束本次循环进入下一次循环，通常不单独使用

break断路：强制终止循环，跳出循环

递推法：注意f1和f2的重复利用

枚举法

打擂台法

素数的算法

函数期中考试不考，教材习题可以做，刷题