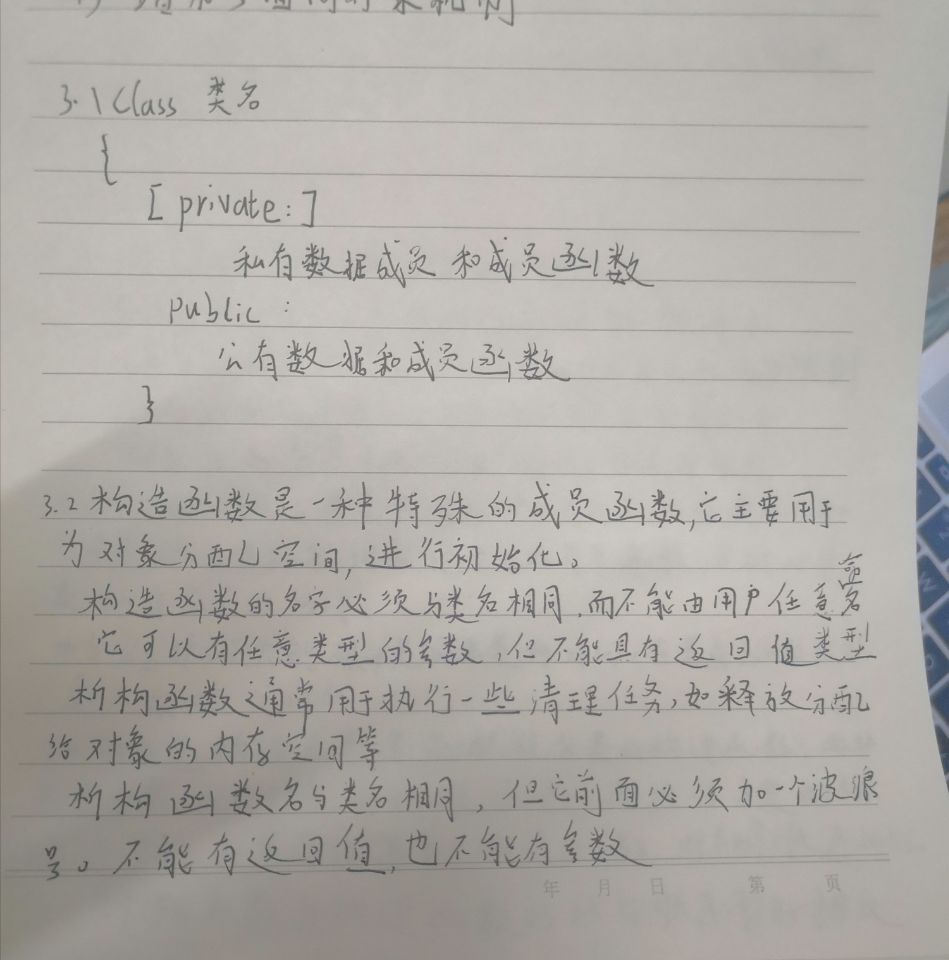
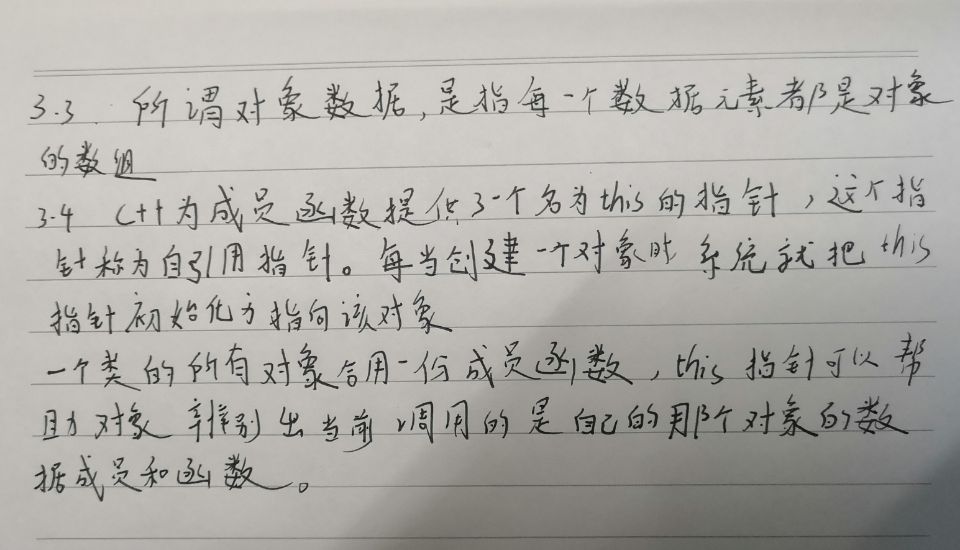
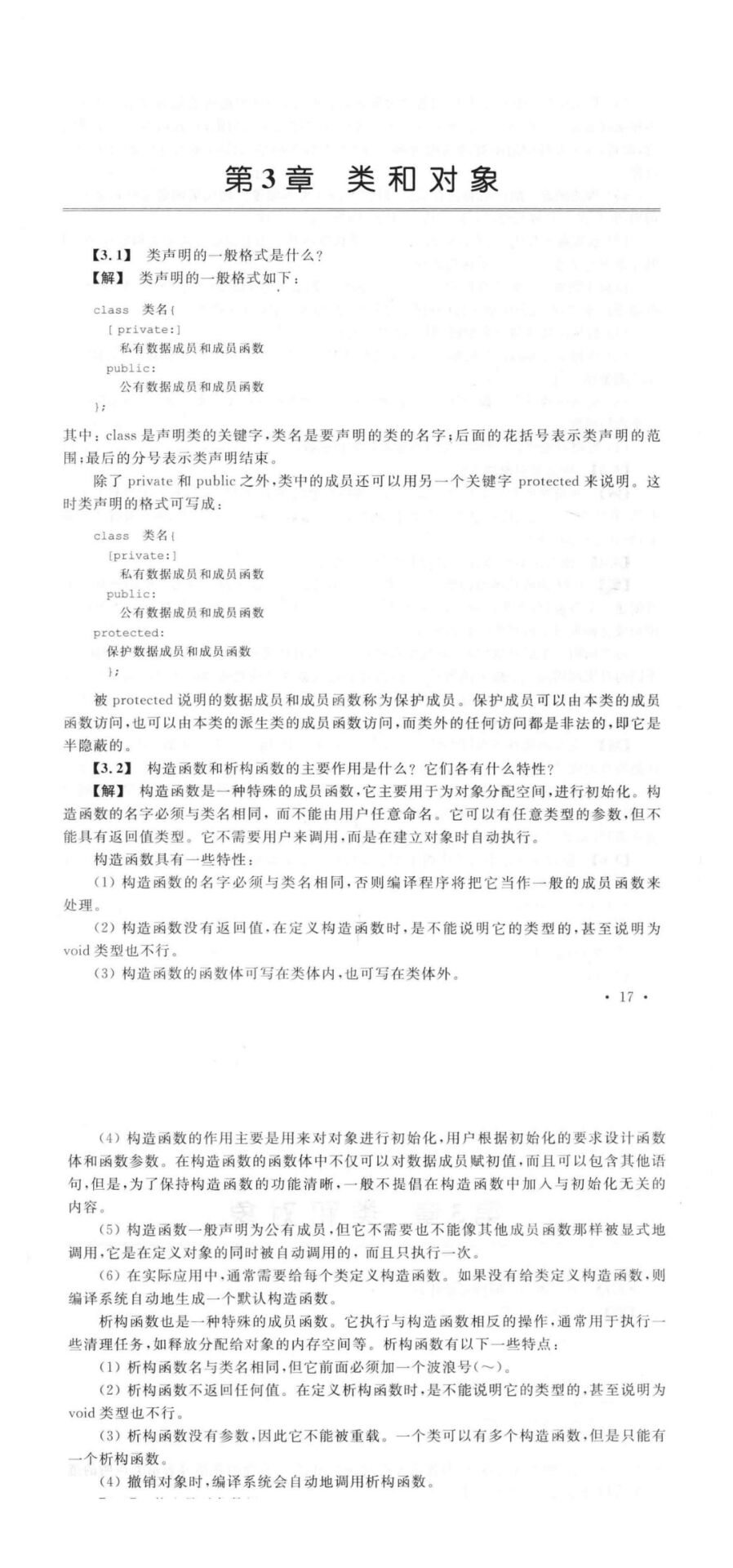
**第三章课后题**

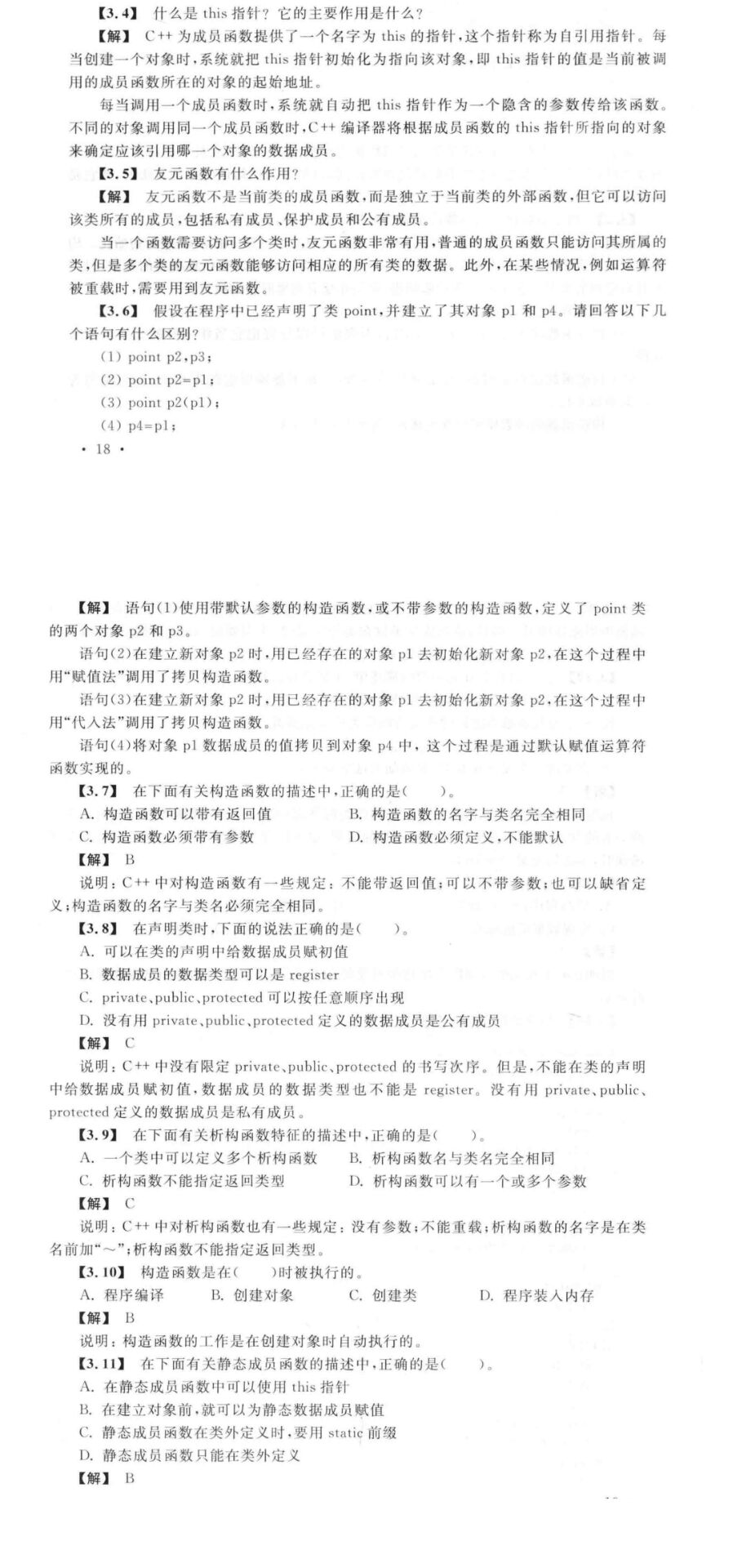
**我的答案：**





**答案：**





**心得感想：**

其实类就好比机械加工领域里的经常听到的开模（这个概念确实不知可以百度下，很easy），比如我们想加工一个齿轮，这个模具是标准件，直接和工厂的师傅说一声就可以加工，但对于我们特殊的工件，如果想成批生产的时候，最好是开模做一个这个特殊工件的模具，然后用这个模具就可以成批的加工一个个的工件。

对于初学C++的我一看到类class的时候，可能都会想：这玩意儿和C的结构体有什么区别么？ 这个问题问的好啊，我就来回答下这个问题：

typedef struct   
{  
int   pos\_flg;   
        char  pid\_pos;   
        float pid\_ang;    
  
pos\_ctl\_t   pos;  
}  
route\_ctl\_t;

这个结构体里可以有各种常规内置类型int char float ，也可以有我们自定义的类型pos\_ctl\_t，我们可以用这个结构体来定义一个结构体变量route\_inf。

但可能写久了的人会想C的结构体里为啥不能写函数呢？于是C++弥补了这个缺憾，并起了个名字叫做类，而用类定义的“变量”也不叫做变量了，而叫做对象

class complex{  
private:  
int a;  
int b;  
  
public://构造函数   
complex(int a = 0, int b = 0) {  
this->a = a;  
this->b = b;  
}  
  
public://重载函数和友元函数   
friend complex operator +(complex &t1, complex &t2);  
  
complex operator-(complex &t){  
complex tmp(this->a - t.a, this->b - t.b);  
return tmp;  
}  
  
public://成员函数   
void comprint() {  
cout<< this->a <<" + " << this->b << "i" <<endl;  
}    
};