

```
//编译器是对成员函数myfunc的解析，是整个A类定义完毕后才开始的；  
//所以，对这个myvar的解析和绑定，是在这个类定义完成后发生的。
```

```
//总结：
```

```
//编译器对 成员函数myfunc的解析，是整个A类定义完毕后才开始，因为只有整个类A定义完毕后，
```

```
//编译器才能看到类A中的myvar，才能根据时机的需要把出现myvar的场合做上述的适当的解释（成员函数中解析成类中的myvar，全局函数中解析成全局的myvar）
```

```
/*A objj:
```

```
  aobj.myvar = 15;
```

```
  aobj.myfunc();*/
```

```
//对于成员函数参数：是在编译器第一次遇到整个类型mytype的时候被决定的，所以，mytype第一次遇到的时候，编译器只看到了  
//typedef string mytype，没有看到类中的typedef in mytype:
```

```
//结论：为了在类中尽早的看到类型mytype，所以这种类型定义语句typedef，一定要挪到类的最开始定义。
```

```
//那后边的成员函数第一次遇到这个类型mytype的时候，它就本看最近碰到的类型的原则来应用最近碰到的类型。
```