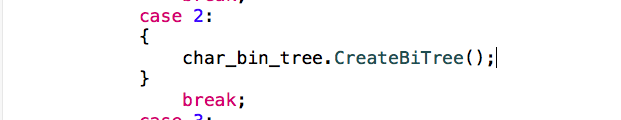
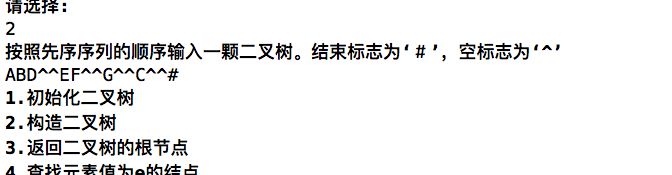
# 跳过 Cin 等待输入的问题

* 最近一直在找bugs

当选择switch为2时，

输入如图



会进入死循环。原本以为是case里执行的函数的问题，后来经老师的排查是由于cin缓存区的问题

程序的输入都建有一个缓冲区，即输入缓冲区。一次输入过程是这样的，当一次键盘输入结束时会将输入的数据存入输入缓冲区，而cin函数直接从输入缓冲区中取数据。正因为cin函数是直接从缓冲区取数据的，所以有时候当缓冲区中有残留数据时，cin函数会直接取得这些残留数据而不会请求键盘输入

解决方法：

当要读入Enter、Space、Tab键时可以用:

1. **cin.get(数组名，长度，结束符)**

其中结束符为可选参数，读入的字符个数最多为（长度-1）个，结束符规定结束字符串读取的字符，默认为ENTER

若要读取字符，直接cin.get(char ch)或ch=cin.get()即可

**读取字符的情况：**

输入结束条件：Enter键

对结束符处理：不丢弃缓冲区中的Enter

cin.get() 与 cin.get(char ch)用于读取字符，他们的使用是相似的，

即：ch=cin.get() 与 cin.get(ch)是等价的。

**2. cin.getline()**

cin.getline(数组名，长度，结束符) 大体与 cin.get(数组名，长度，结束符)类似。

区别在于：

cin.get()当输入的字符串超长时，不会引起cin函数的错误，后面的cin操作会继续执行，只是直接从缓冲区中取数据。但是cin.getline()当输入超长时，会引起cin函数的错误，后面的cin操作将不再执行。

当不需要读入Enter、Space、Tab键时可以用:

cin.ignore（）和 cin.sync

区别如图，引用自http://www.cnblogs.com/silentNight/p/5452814.html

