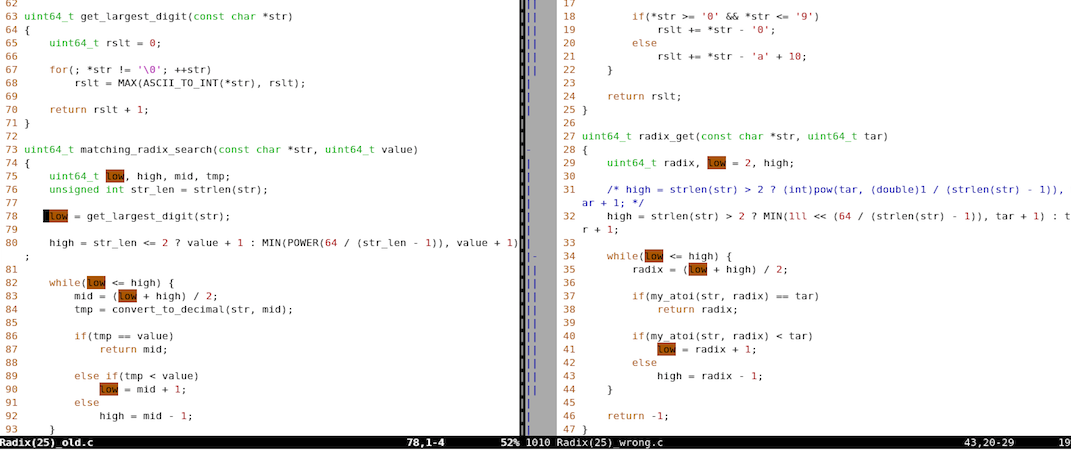
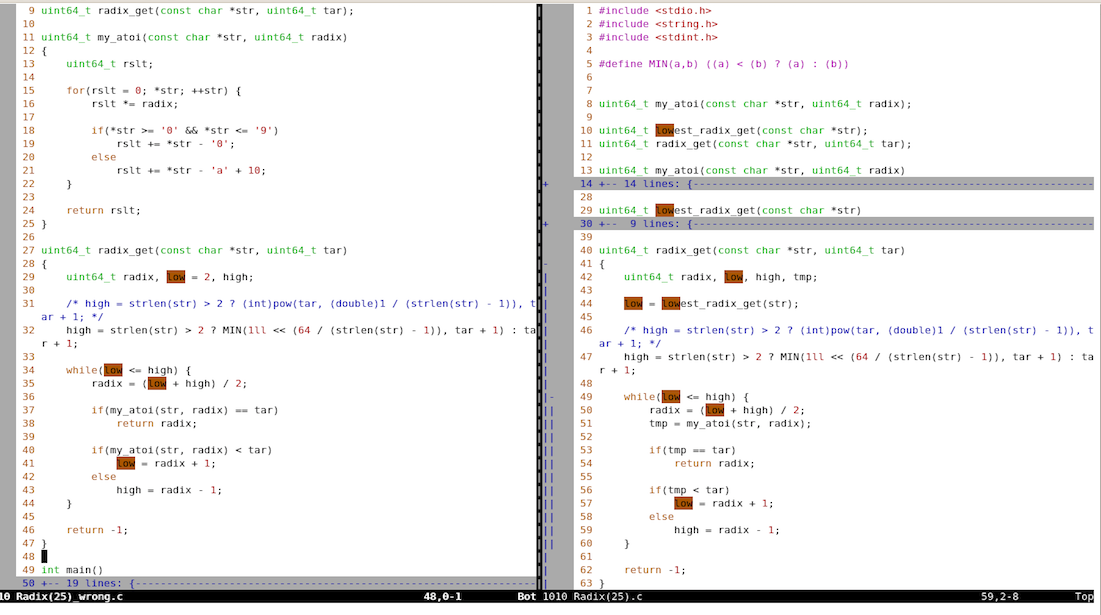
依据对题意的分析，我初试版本的代码仅紧确了最小根r的上界，没有紧确其下界。开始嫌麻烦，以为即使下界从2开始寻找，二分查找后也能找到匹配的根，不紧确的后果只不过是查找范围稍大了一点，结果有2个case过不去。

开始调试的时候百思不得其解，以为是其他环节出了问题，几经排查还是没想到是二分查找的下界不正确。于是翻看以前写过的code，即\*old.c，发现了倪端。



\*old.c是可以AC的，这个版本的核心处理比我新写的\*wrong.c多了对low的处理。于是我在\*wrong.c的基础上追加了紧确下界的处理，这个版本是可以通过的。



当然两个case不通过的原因，就是少了对下界的紧确处理。直观地看就是return的结果不正确。

对于找不到匹配根的情形，都返回-1，这个处理时没有问题的，于是问题就精确到了\*wrong.c L38：return radix。

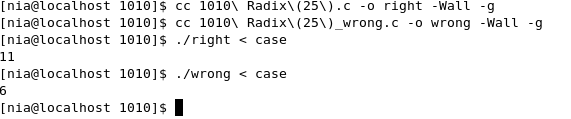
而radix完全由low和high计算而来，low的不紧确会影响radix。换言之，就是取了一个不正确的low，计算出radix的值也不正确，程序出错。

由上述推断，以及串数转换的函数处理可知，当radix值小于串上最大位的值时(不合法)，依据my\_atoi的规则是可以算出结果的。假设串只有一位x，且该值大于radix，my\_atoi算出的结果仍是x。

于是构造测试用例验证：

1010 a 1 2

正确版和错误版的运行结果如下：



综上，对于此bug出现的原因分析完毕。

PS：旧版本题解\*old.c，出bug的新解\*.c，以及正确新解\*.c都保留在当前目录下。