**atoi**

**2018K8009918002 陈彦帆**

64位汇编程序 atoi.S

C测试程序 main.c

1. 编译运行命令

（如果运行失败请联系我）

gcc -c main.c

gcc -c atoi.S

gcc atoi.o main.o -o main

./main

2. 正确性测试及运行结果

选取十组测试数据如下

    char s1[]=" +12345";

    char s2[]="+asd777";

    char s3[]="-999999999999asd";

    char s4[]="2147483647+";

    char s5[]="+2147483648+ss";

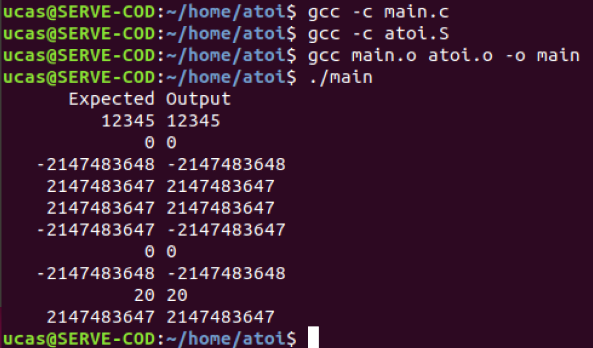
    char s6[]=" -2147483647 d";

    char s7[]="00ww00";

    char s8[]="          -2147483648";

    char s9[]=" 020 3";

    char s10[]=" 9999999997";

运行结果正确：

3. 算法描述

本程序所有临时变量都放在寄存器中，无需建立栈帧。

搬运：每次从输入地址搬运一个字符b并零扩展到寄存器rcx。并将地址加一。

(1)判断空格：进行一步搬运。若字符为空格，继续判断空格，否则进入下一步。

(2)清零寄存器rax用于存储结果sum。清零寄存器dl用于存储符号位。判断字符是否为’+’，若是，搬运一次，进入(4)。否则进入(3)

(3)判断字符是否为’-’, 若是，将dl设为1，搬运一次，进入(4)，否则直接进入(4)

(4)判断字符b是否在’0’-‘9’之间。若不是，进入(6)。若是，sum=sum\*10+(b-‘0’)，可用两条lea指令完成。但是lea指令不设置溢出位，因此需要手动判断溢出，方法是将sum与INT\_MAX进行无符号比较，若sum更大，则溢出。因为sum是64位而返回值只要求32位，故这种判断方法是成立的。若溢出，进入(5)，否则继续(4)。

(5) 溢出：若dl为0，返回INT\_MAX，否则返回INT\_MIN。

(6) 若dl为0，返回；否则将结果取反后返回。