

**实验报告**



**题目： Linux环境和GCC工具链**

**班 级： 2020211310**

**学 号： 2020211502**

**姓 名： 王小龙**

**学 院： 计算机学院**

**2021年 11月 3 日**

一、实验目的

1. 熟悉linux操作的基本操作；
2. 掌握gcc编译方法；
3. 掌握gdb的调试工具使用；
4. 掌握objdump反汇编工具使用；
5. 熟悉理解反汇编程序（对照源程序与objdump生成的汇编程序）。
6. 实验环境

列举你所使用的软件工具

1. xshell
2. Dev-C++

三、实验内容

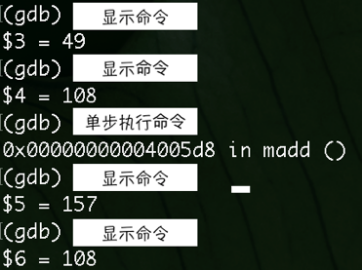
现有int型数组a[i]=i-50,b[i]=i+y，其中y取自于学生本人学号2019211x\*y的个位。登录bupt1服务器，在linux环境下使用vi编辑器编写C语言源程序，完成数组a+b的功能，规定数组长度为100，函数名为madd（），数组a，b均定义在函数内，采用gcc编译该程序（使用-g选项，但不使用优化选项），

1. 使用objdump工具生成汇编程序，找到madd函数的汇编程序，给出截图；
2. 用gdb进行调试，练习如下gdb命令，给出截图；

gdb、file、kill、quit、break、delete、clear、info break、run、continue、nexti、stepi、disassemble、list、print、x、info reg、watch

1. 找到a[i]+b[i]对应的汇编指令，指出a[i]和b[i]位于哪个寄存器中，给出截图；
2. 使用单步指令及gdb相关命令，显示a[xy]+b[xy]对应的汇编指令执行前后操作数寄存器十进制和十六进制的值，其中x，y取自于学生本人学号2019211x\*y的百位和个位。

学号2019211999，a[99]+b[99]单步执行前后的参考截图如下（实际命令未显示出）：

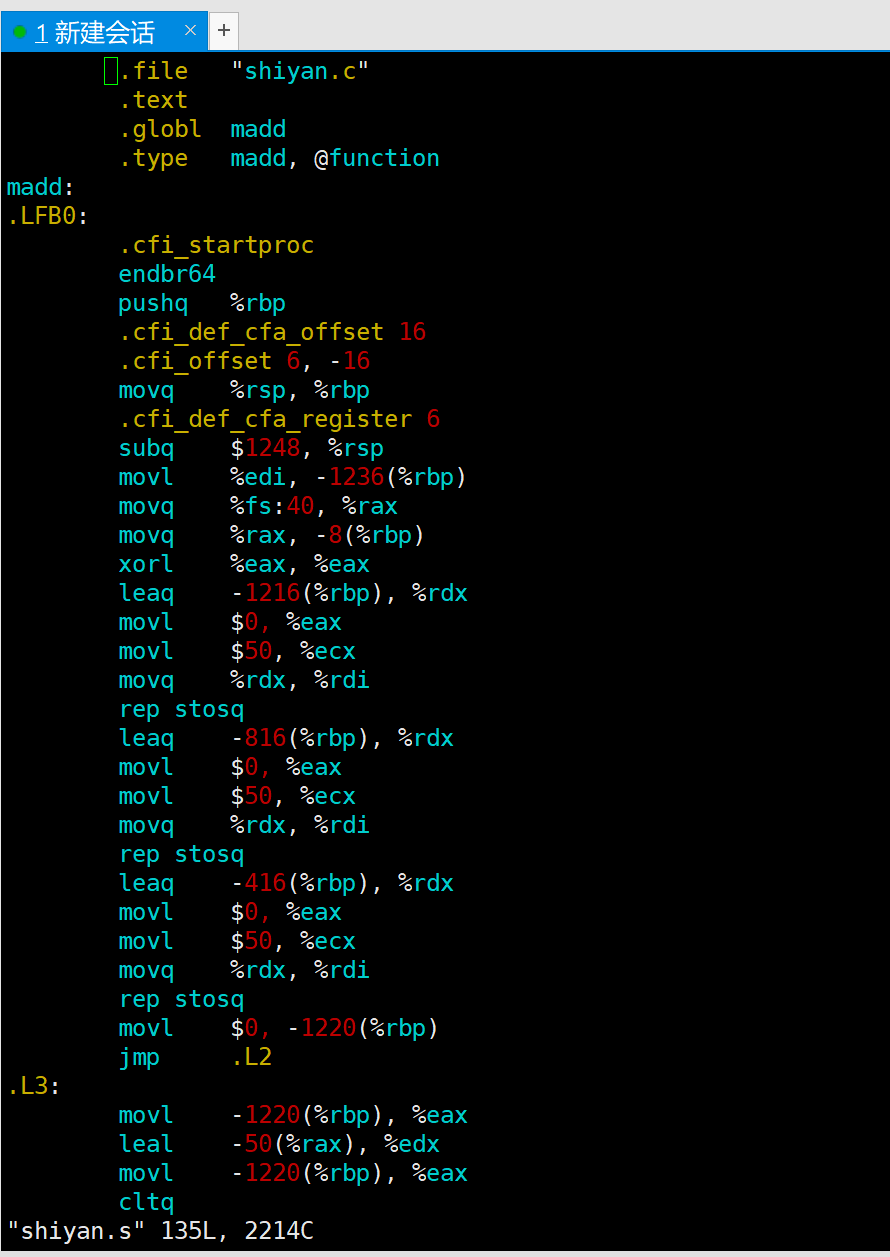


四、实验步骤及实验分析

操作步骤：

1. 把shiyan.c输入到vi。输入gcc -S shiyan.c -o shiyan.s得到汇编程序。

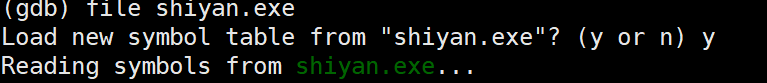
输入vi shiyan.s可看到汇编程序，截图如下：

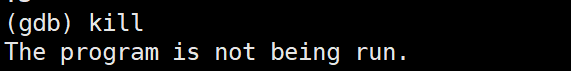


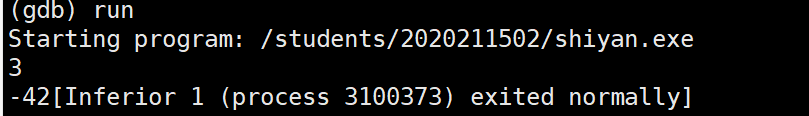
1. 输入gcc -g shiyan.c -o shiyan.exe 得到可执行文件。

输入gdb shiyan.exe进入gdb,

各命令运行截图：

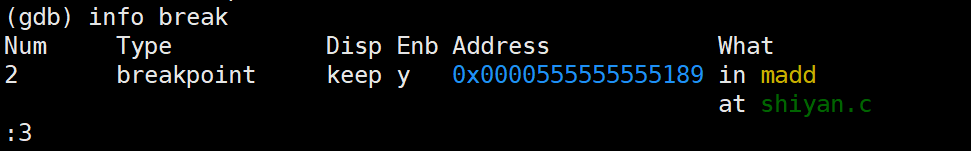


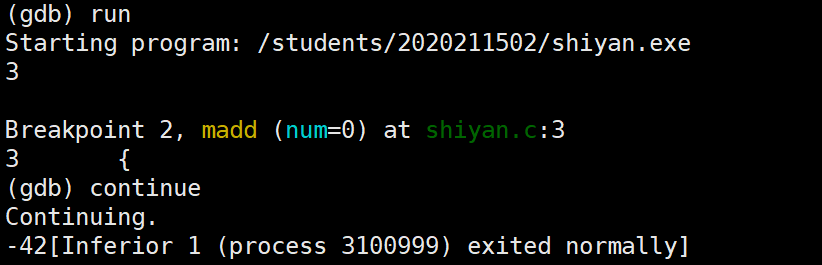


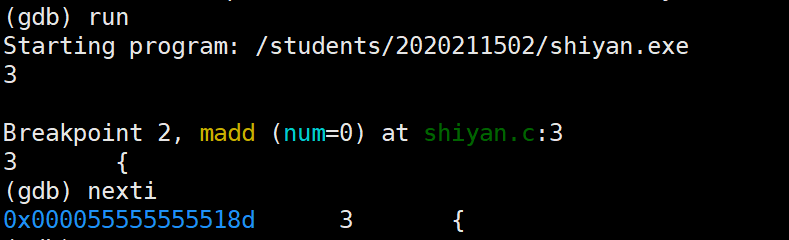


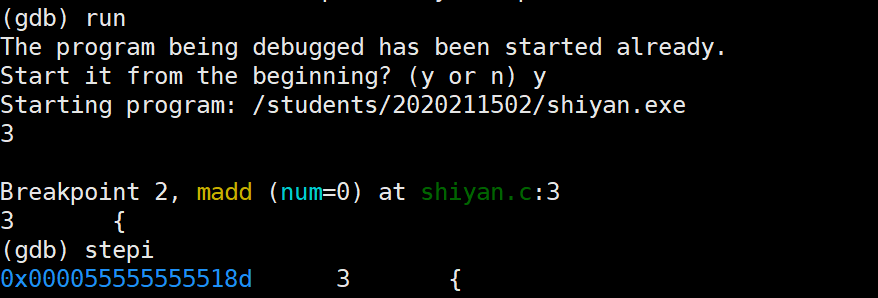
IMG_256

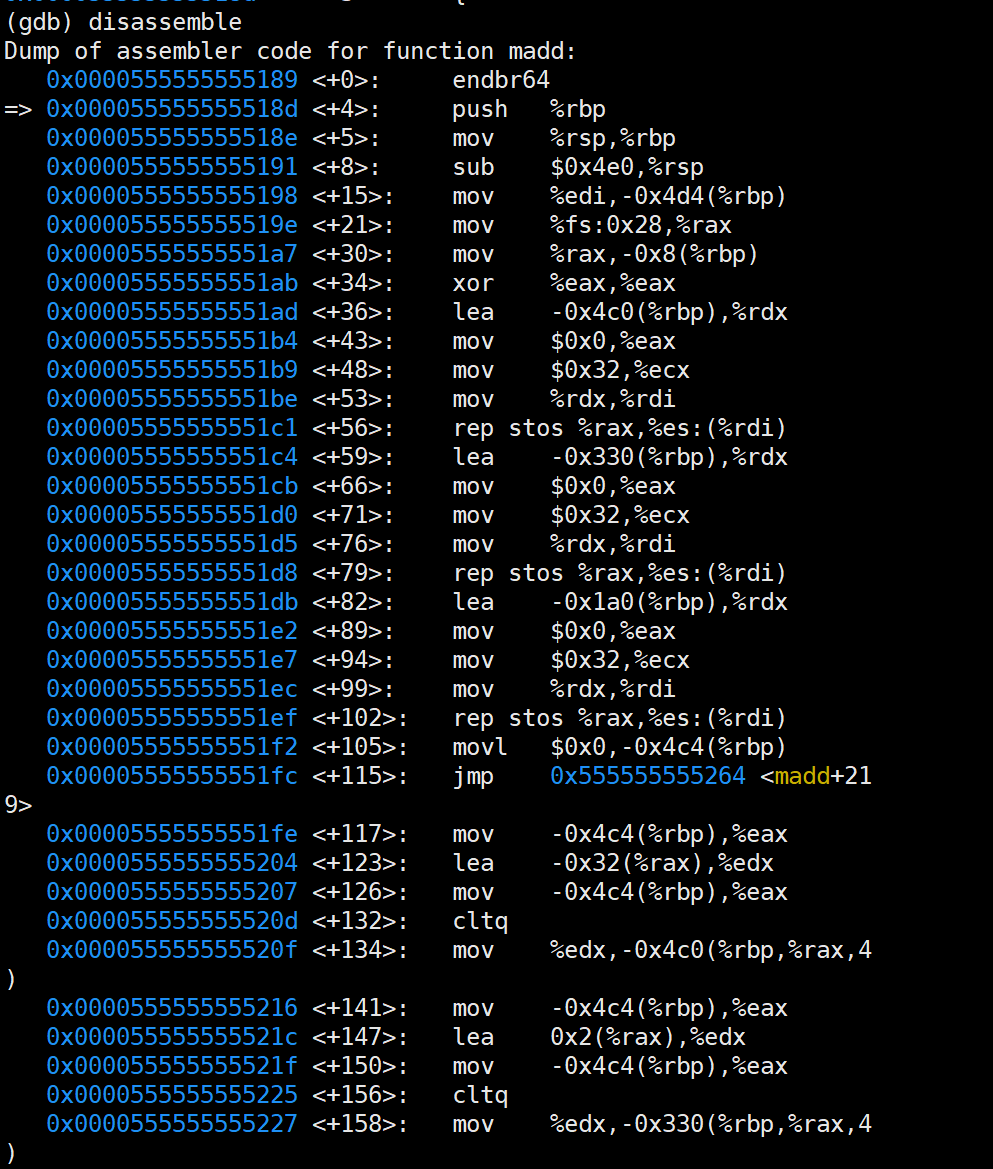
IMG_256

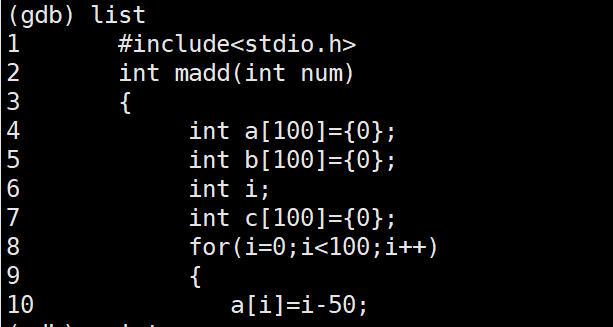


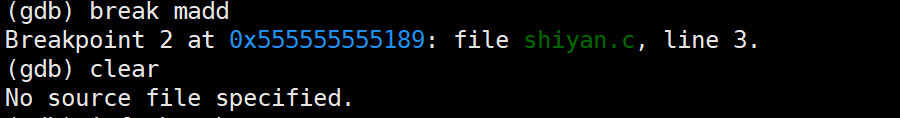


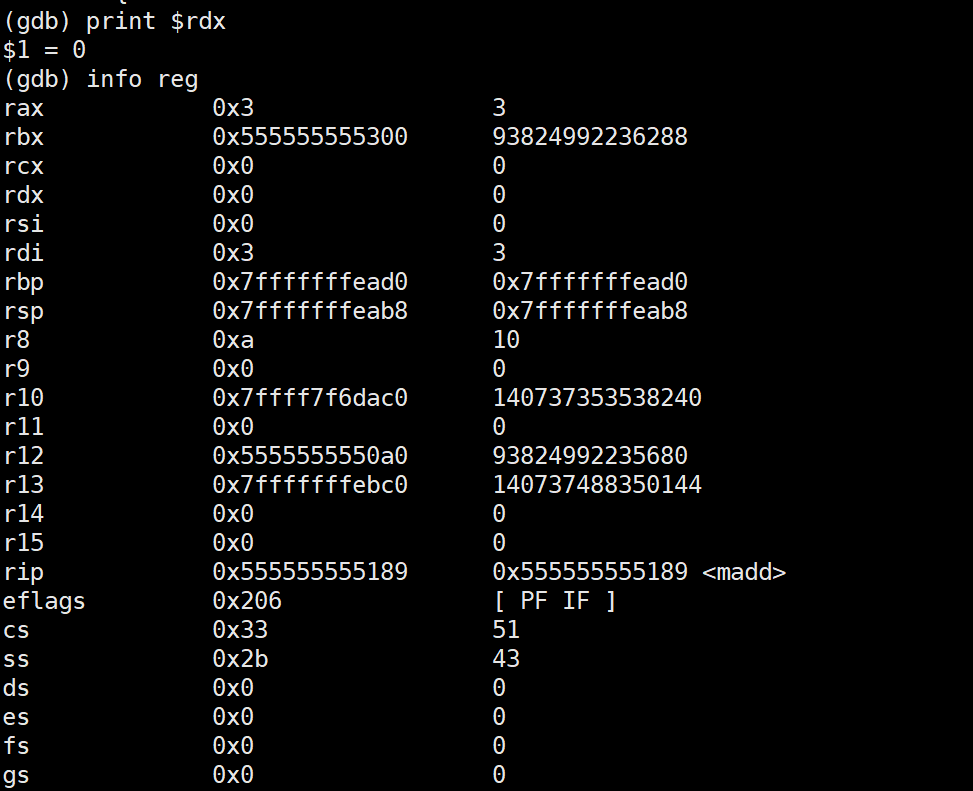


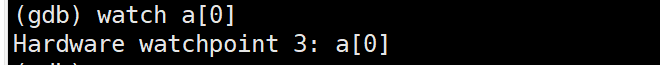




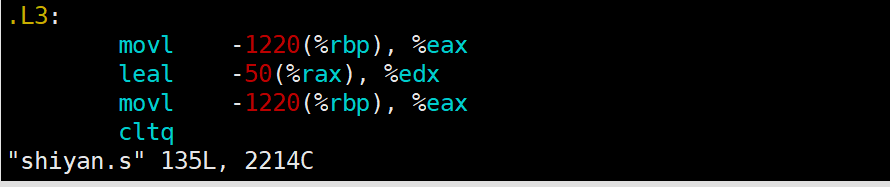




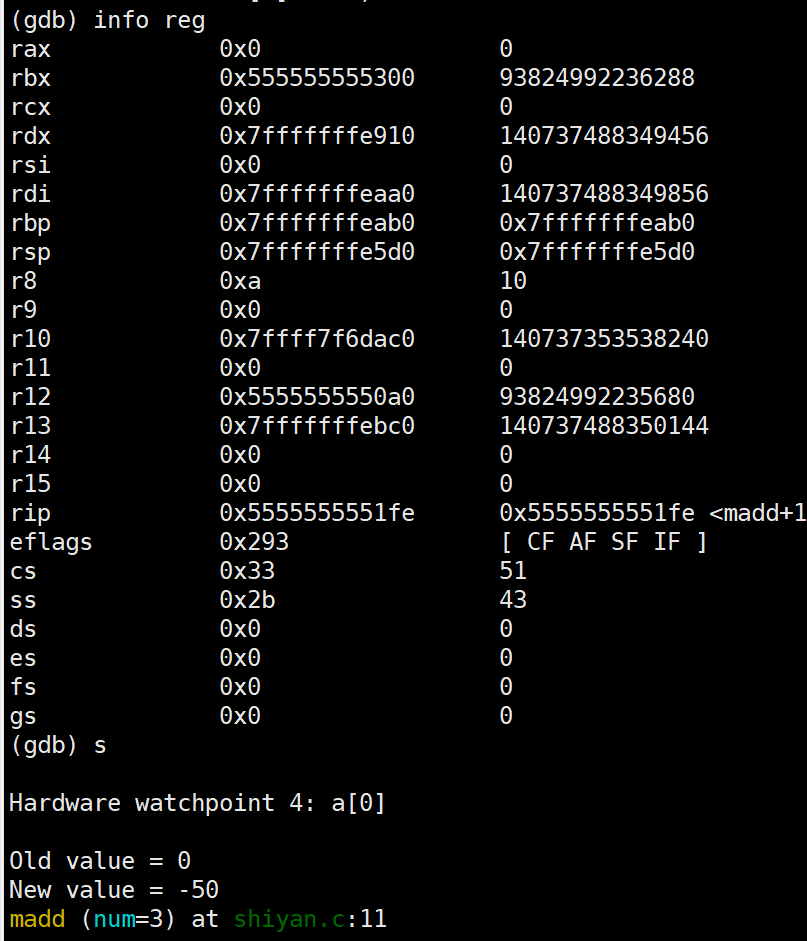


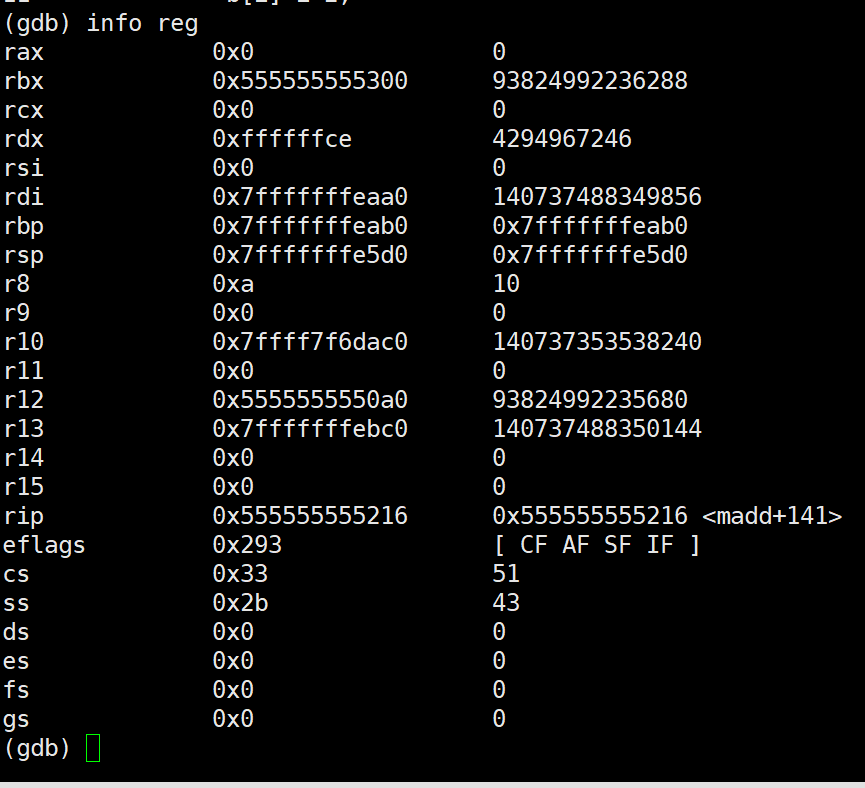


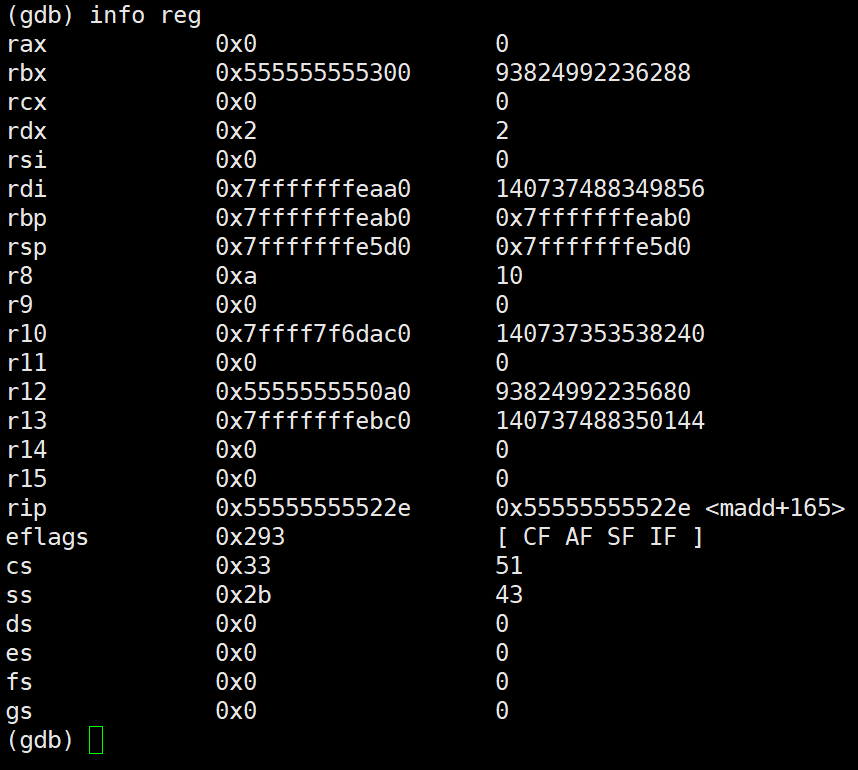
1. a[i]+b[i]对应的汇编指令截图：



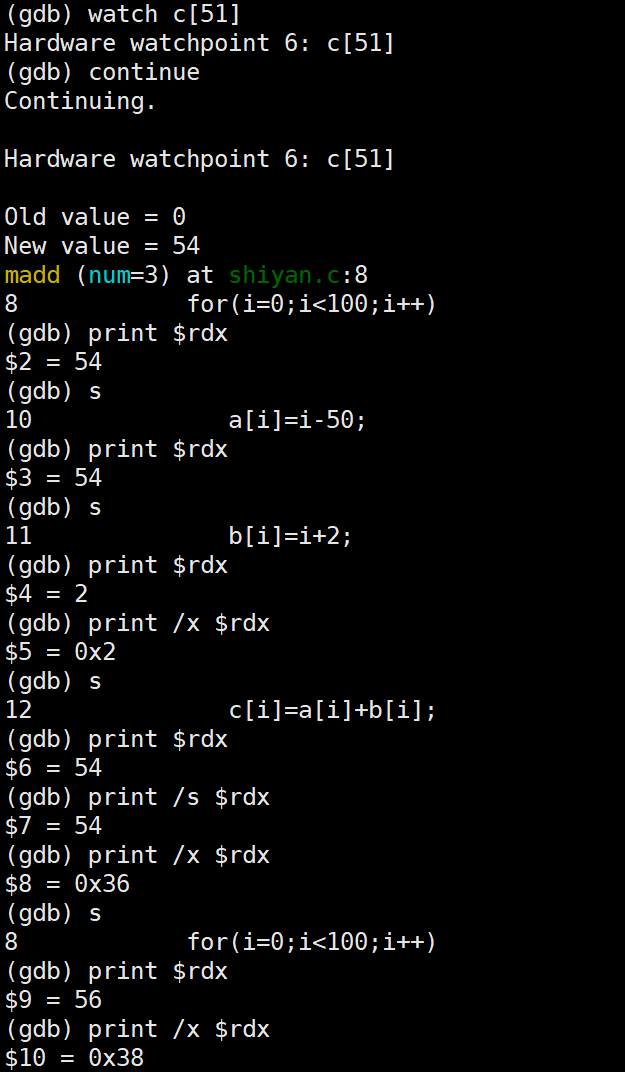
根据如下3张截图可知a[i]位于rdx寄存器，b[i]位于rdx寄存器：







1. 使用print $rdx和print /x $rdx等指令，显示a[xy]+b[xy]对应的汇编指令执行前后操作数寄存器十进制和十六进制的值如下截图所示：



五、总结体会

总结心得：

1. 遇到的问题：对各操作结果不熟悉，出现错误后难以发现问题。
2. 如何解决的：多次练习和查找资料得以解决。