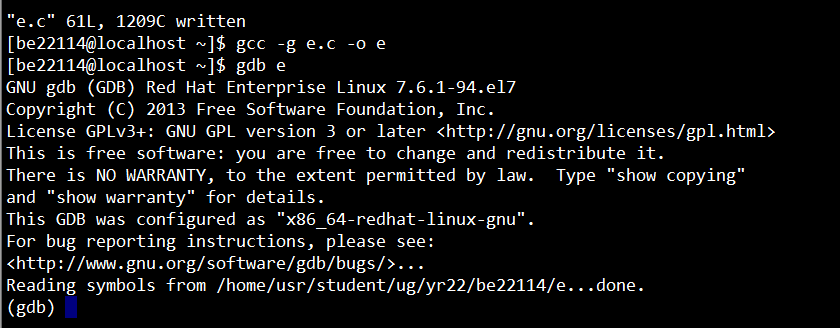
**ASSIGNMENT-3 SE LAB**

**NAME: SAMUDRA ROY ROLL:002211001114 SE:A3**

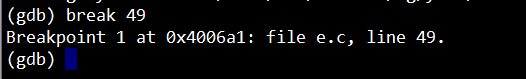
**1. Consider the program in Assign3.It is a simple**

**state machine.**

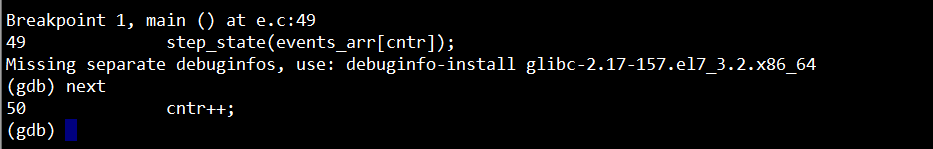


**a. Put a breakpoint in line 49**

ANS: break 49



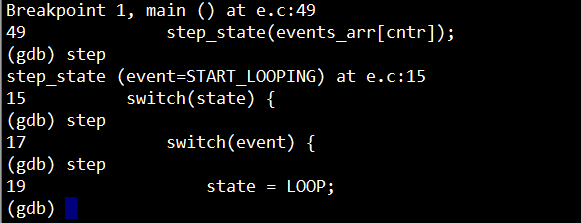
**b. Try next command**



**c. How will you get inside the function without**

**using breakpoint?**

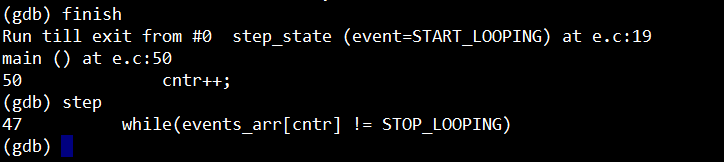
**Ans:** step=> Runs the next instruction, not line. If the current instruction is setting a variable, it is the same as next. If it’s a function, it will jump into the function, execute the first statement, then pause.



**d. How will you come out the of the function**

**without using next and continue?**

**Ans:** finish => Finishes executing the current function, then pause (also called step out).



**2. Consider the program in Assign4 .It is also a simple**

**state machine.If you provide user id and password**

**properly account details will be displayed. The basic**

**rule is user id should be positive and less than 20**

**.password is userid \*b1000 .The loop will terminate**

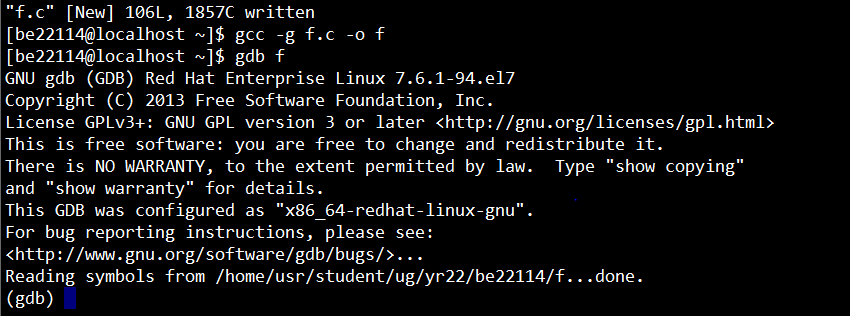
**after 10 iteration. It works fine if you provide valid user**

**id and password.It works fine for invalid userid. But it**

**goes to infiniteloop for invalid password.Run the**

**program .It goes into infinite loop.you need to kill the**

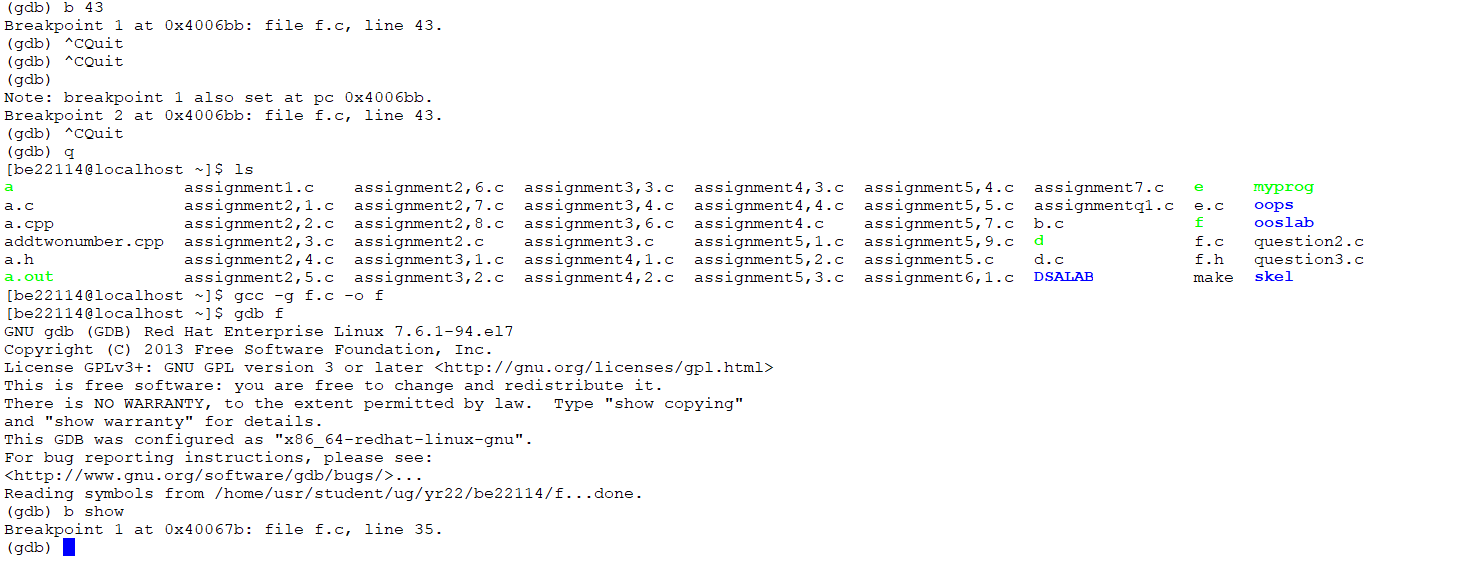
**program by [ctrl^c]**



**a. Set a suitable breakpoint in gdb in the routine**

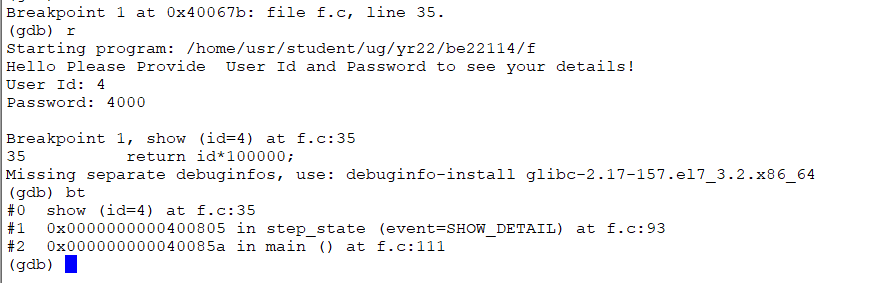
**show.give valid input and run :**

break 43



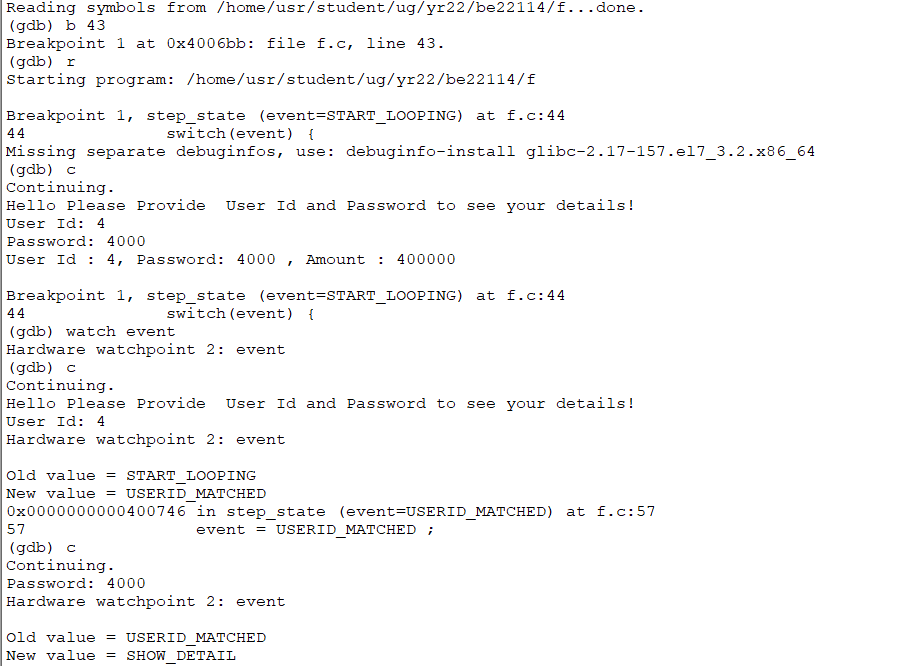
**b. How you can see the call stack of the routine.**

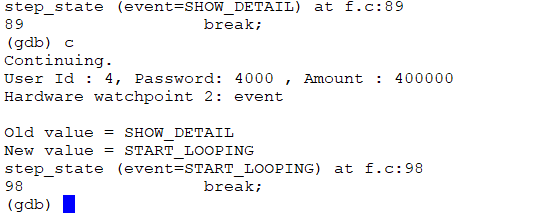
**Ans:** bt

****

**c. Which commands will help you to see each value**

**change of variable “event”?**

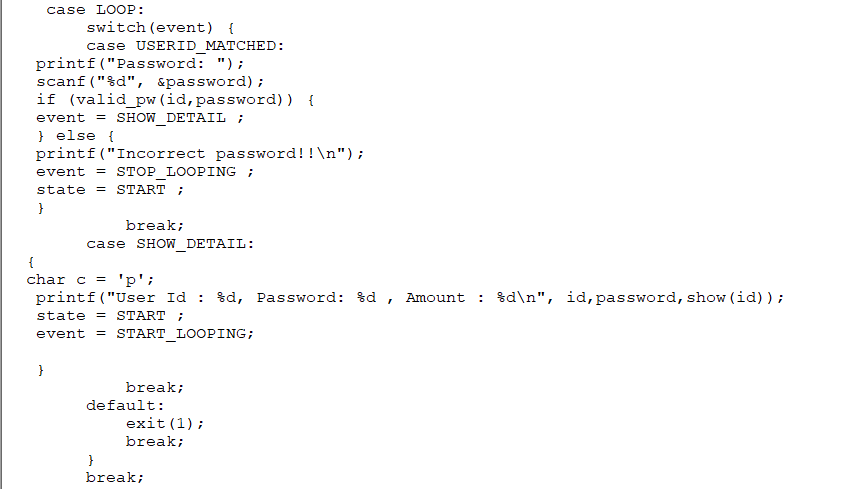
**Ans**: Watch event



**d. Correct the program so that it doesn’t go to infinite loop for wrong password. Rather main iteration restarts . [follow the value change path of event for wrong password]**

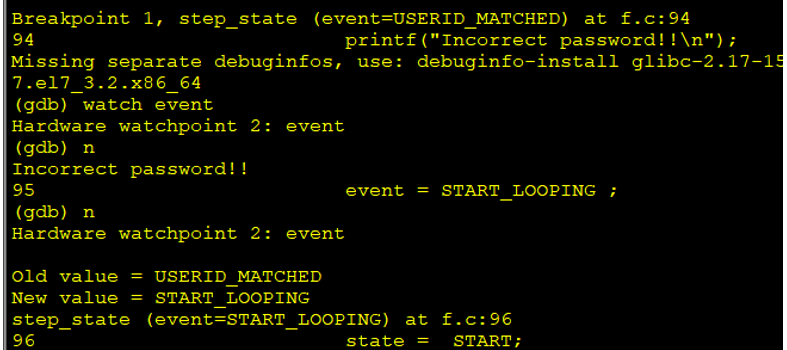
**ans:**

Changed *event = STOP\_LOOPING* to *event=START\_LOOPING* at line 94/95

****

**Explore the commands found for 5c to see/use**

**content of a pointer**

****