Assignments-2 on gdb

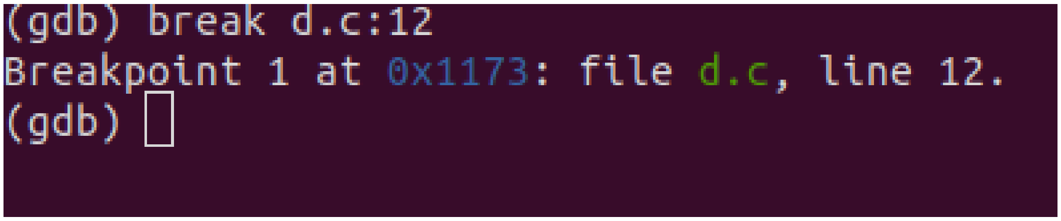
Name:- ATRIJ ROY

Roll no:- 002311001086

Class:- IT UG2 A3

a> Consider the program in folder assign2.

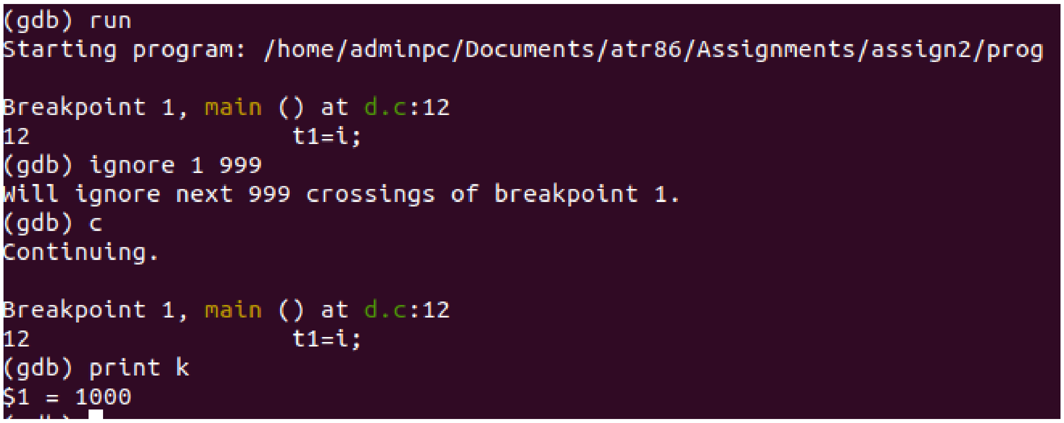
a. Put a breakpoint in the 1st executable line of the innermost loop.



```
(gdb) break d.c:12
Breakpoint 1 at 0x1173: file d.c, line 12.
(gdb) 
```

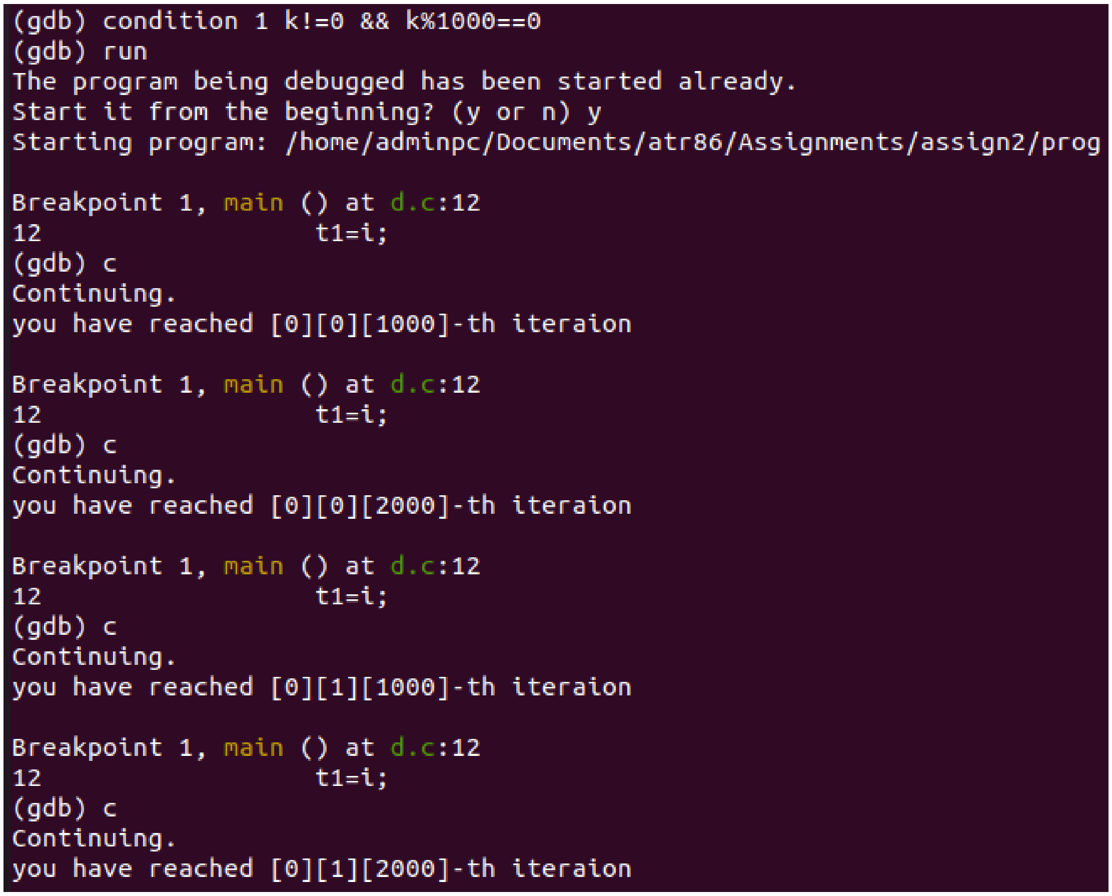
b. If you run and continue ,how many times it is supposed to stop at breakpoint 1?
10*200*3000=6000000 times

c. How will you continue so that it stops at the 1000th iteration of the innermost loop ?



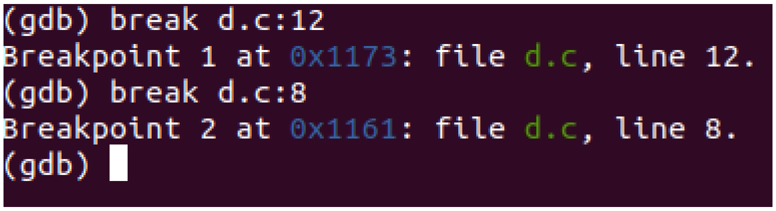
```
(gdb) run
Starting program: /home/adminpc/Documents/atr86/Assignments/assign2/prog

Breakpoint 1, main () at d.c:12
12                    t1=i;
(gdb) ignore 1 999
Will ignore next 999 crossings of breakpoint 1.
(gdb) c
Continuing.

Breakpoint 1, main () at d.c:12
12                    t1=i;
(gdb) print k
$1 = 1000
(gdb) 
```

d. How you can condition your breakpoint , so that the loop stops at every 1000th iteration of innermost loop ?



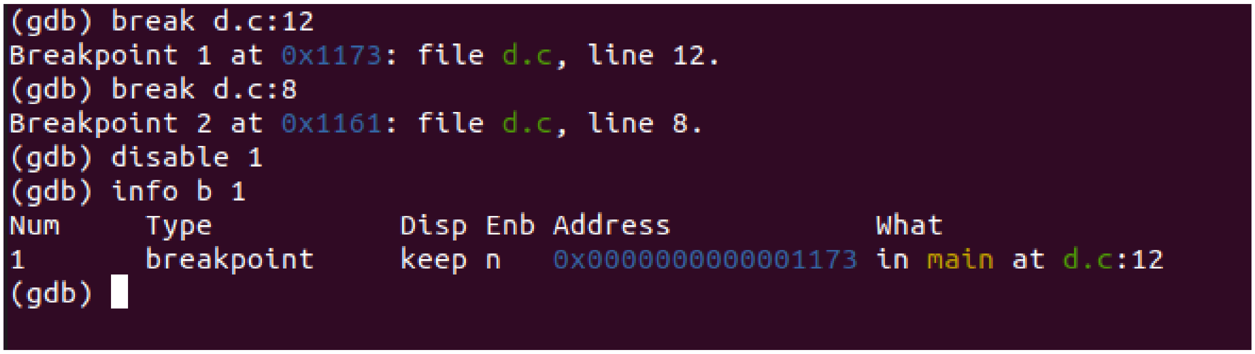
```
(gdb) condition 1 k!=0 && k%1000==0
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/adminpc/Documents/atr86/Assignments/assign2/prog

Breakpoint 1, main () at d.c:12
12                      t1=i;
(gdb) c
Continuing.
you have reached [0][0][1000]-th iteraion

Breakpoint 1, main () at d.c:12
12                      t1=i;
(gdb) c
Continuing.
you have reached [0][0][2000]-th iteraion

Breakpoint 1, main () at d.c:12
12                      t1=i;
(gdb) c
Continuing.
you have reached [0][1][1000]-th iteraion

Breakpoint 1, main () at d.c:12
12                      t1=i;
(gdb) c
Continuing.
you have reached [0][1][2000]-th iteraion
```

e. Put a breakpoint in the 1st line of the outermost loop.



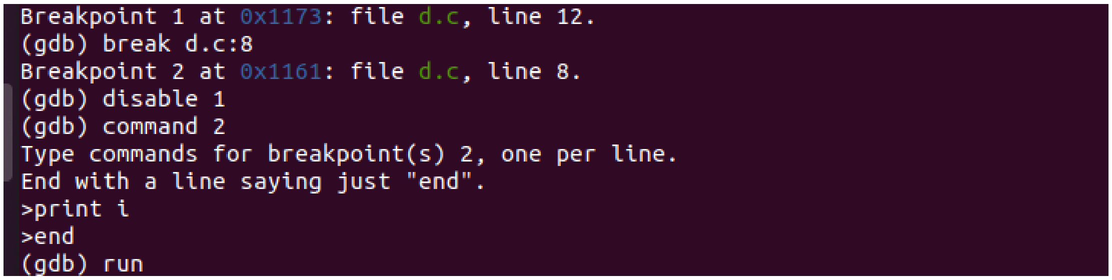
```
(gdb) break d.c:12
Breakpoint 1 at 0x1173: file d.c, line 12.
(gdb) break d.c:8
Breakpoint 2 at 0x1161: file d.c, line 8.
(gdb)
```

f. Disable breakpoint “1”

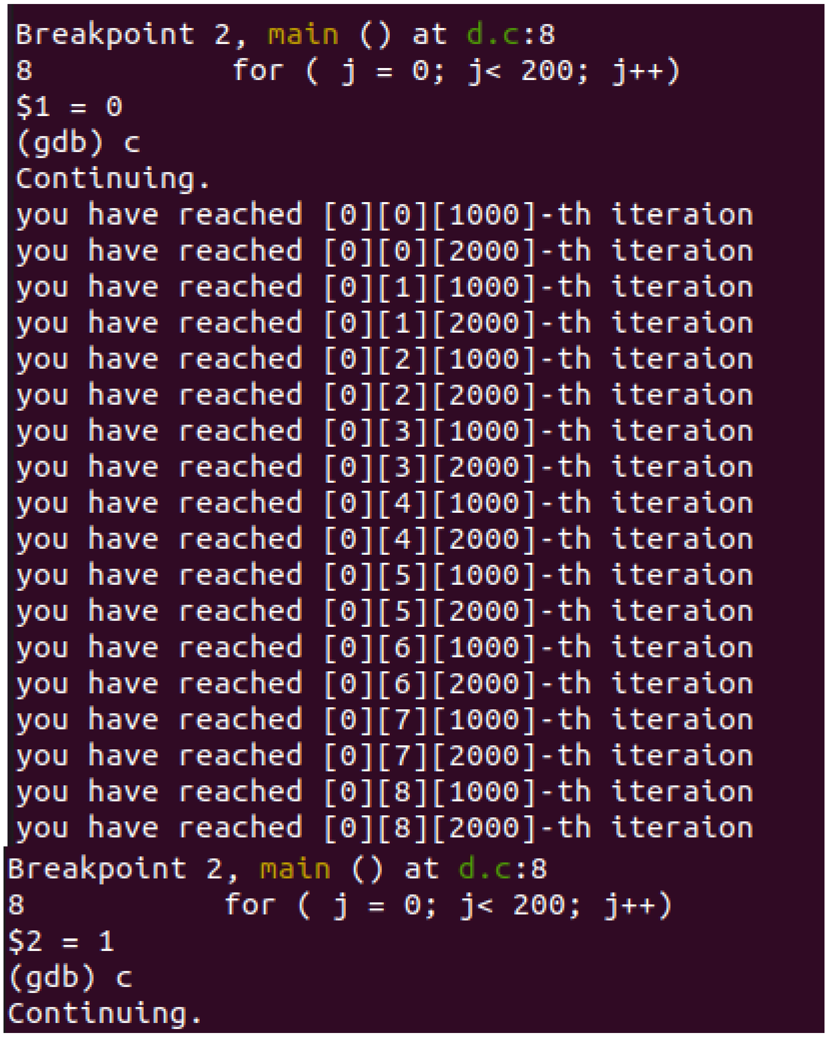


```
(gdb) break d.c:12
Breakpoint 1 at 0x1173: file d.c, line 12.
(gdb) break d.c:8
Breakpoint 2 at 0x1161: file d.c, line 8.
(gdb) disable 1
(gdb) info b 1
Num     Type           Disp Enb Address            What
1       breakpoint     keep n   0x0000000000001173 in main at d.c:12
(gdb) 
```

g. Add a command to breakpoint 2 so that it prints the value of “i” at each hit.

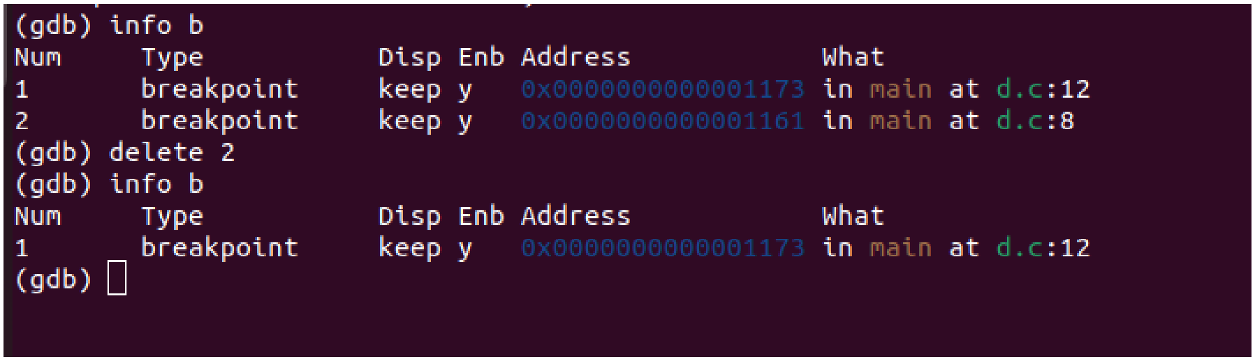


```
Breakpoint 1 at 0x1173: file d.c, line 12.
(gdb) break d.c:8
Breakpoint 2 at 0x1161: file d.c, line 8.
(gdb) disable 1
(gdb) command 2
Type commands for breakpoint(s) 2, one per line.
End with a line saying just "end".
>print i
>end
(gdb) run
```



```
Breakpoint 2, main () at d.c:8
8           for ( j = 0; j< 200; j++)
$1 = 0
(gdb) c
Continuing.
you have reached [0][0][1000]-th iteraion
you have reached [0][0][2000]-th iteraion
you have reached [0][1][1000]-th iteraion
you have reached [0][1][2000]-th iteraion
you have reached [0][2][1000]-th iteraion
you have reached [0][2][2000]-th iteraion
you have reached [0][3][1000]-th iteraion
you have reached [0][3][2000]-th iteraion
you have reached [0][4][1000]-th iteraion
you have reached [0][4][2000]-th iteraion
you have reached [0][5][1000]-th iteraion
you have reached [0][5][2000]-th iteraion
you have reached [0][6][1000]-th iteraion
you have reached [0][6][2000]-th iteraion
you have reached [0][7][1000]-th iteraion
you have reached [0][7][2000]-th iteraion
you have reached [0][8][1000]-th iteraion
you have reached [0][8][2000]-th iteraion
Breakpoint 2, main () at d.c:8
8           for ( j = 0; j< 200; j++)
$2 = 1
(gdb) c
Continuing.
```

h. Delete breakpoint 2.



```
(gdb) info b
Num     Type           Disp Enb Address            What
1       breakpoint     keep y   0x0000000000001173 in main at d.c:12
2       breakpoint     keep y   0x0000000000001161 in main at d.c:8
(gdb) delete 2
(gdb) info b
Num     Type           Disp Enb Address            What
1       breakpoint     keep y   0x0000000000001173 in main at d.c:12
(gdb) 
```