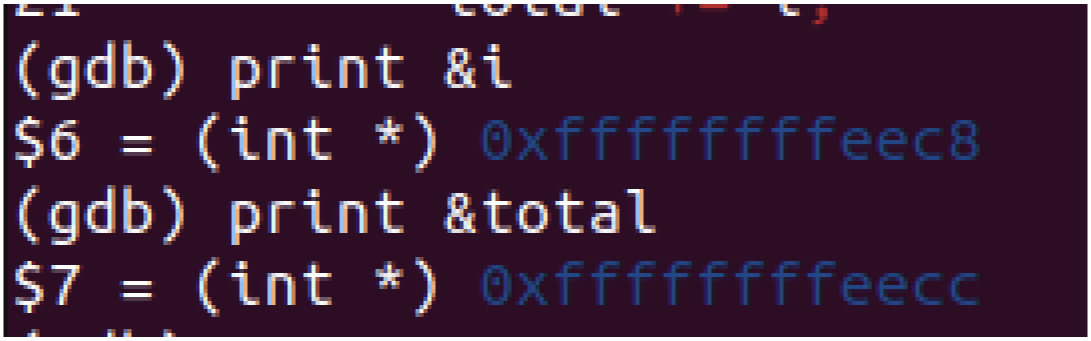
1)



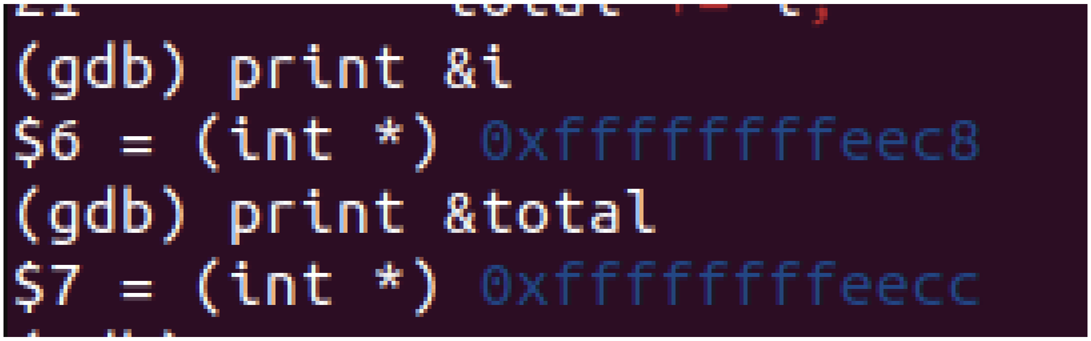
```
(gdb) print &i
$6 = (int *) 0xffffffffeec8
(gdb) print &total
$7 = (int *) 0xffffffffeecc
```

0xffffffffeec8 < 0xffffffffeecc ->&i < & total

Since the stack goes from higher addresses to lower addresses, the total is on top and i is on the bottom.

So, this compiler, adds y closer to the top of the stack.

2)



```
(gdb) print &i
$6 = (int *) 0xffffffffeec8
(gdb) print &total
$7 = (int *) 0xffffffffeecc
```

3) Just by observing memory addresses of i and total, new variables get added to higher memory addresses.