

```
[izidor@myfedora20 ~]$ cat test.cpp
```

```
int main()
{
```

```
int l = 22;
int y = -2;
int c = l + y;
```

```
return 0;
```

```
}
[izidor@myfedora20 ~]
```

```
>
```

```
> ^C
```

```
[izidor@myfedora20 ~]$ cc -g -o0 test.c -o tt
```

```
[izidor@myfedora20 ~]$ gdb tt
```

```
GNU gdb (GDB) Fedora 7.7.1-21.fc20
```

```
Copyright (C) 2014 Free Software Foundation, Inc.
```

```
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
```

```
This is free software: you are free to change and redistribute it.
```

```
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
```

```
This GDB was configured as "x86_64-redhat-linux-gnu".
```

```
Type "show configuration" for configuration details.
```

```
For bug reporting instructions, please see:
```

```
<http://www.gnu.org/software/gdb/bugs/>.
```

```
Find the GDB manual and other documentation resources online at:
```

```
<http://www.gnu.org/software/gdb/documentation/>.
```

```
For help, type "help".
```

```
Type "apropos word" to search for commands related to "word"...
```

```
Reading symbols from tt...done.
```

```
(gdb) break main
```

```
Breakpoint 1 at 0x4004f4: file test.c, line 7.
```

```
(gdb) run
```

```
Starting program: /home/izidor/tt
```

```
Breakpoint 1, main () at test.c:7
```

```
7      int a = 5;
```

```
Missing separate debuginfos, use: debuginfo-install glibc-2.18-19.fc20.x86_64
```

```
(gdb) next 2
```

```
9      int c = -3;
```

```
(gdb) disassemble
```

```
Dump of assembler code for function main:
```

```
0x00000000004004f0 <+0>:      push  %rbp
0x00000000004004f1 <+1>:      mov   %rsp,%rbp
0x00000000004004f4 <+4>:      movl  $0x5,-0x4(%rbp)
0x00000000004004fb <+11>:     movl  $0xffffffff,-0x8(%rbp)
=> 0x0000000000400502 <+18>:     movl  $0xffffffff,-0xc(%rbp)
```

```

0x0000000000400509 <+25>:  mov  -0x8(%rbp),%eax
0x000000000040050c <+28>:  mov  -0x4(%rbp),%edx
0x000000000040050f <+31>:  add  %eax,%edx
0x0000000000400511 <+33>:  mov  -0xc(%rbp),%eax
0x0000000000400514 <+36>:  add  %eax,%edx
0x0000000000400516 <+38>:  mov  0x200b10(%rip),%eax    # 0x60102c <x.1723>
0x000000000040051c <+44>:  add  %edx,%eax
0x000000000040051e <+46>:  mov  %eax,-0x10(%rbp)
0x0000000000400521 <+49>:  mov  $0x0,%eax
0x0000000000400526 <+54>:  pop  %rbp
0x0000000000400527 <+55>:  retq

```

End of assembler dump.

(gdb) display/i \$pc

1: x/i \$pc

=> 0x400502 <main+18>: movl \$0xffffffff,-0xc(%rbp)

(gdb) displa