

GDB: The GNU Project Debugger

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

Gnu Debugger

Compile o seu programa com a opção `-g`

```
gcc -o prog prog.c -g -Wall -ansi -pedantic -O2
```

Para executar o seu programa faça

```
meu_prompt> gdb prog
```

```
GNU gdb 6.3-debian
```

```
Copyright 2004 Free Software Foundation, Inc.
```

```
GDB is free software, covered by the GNU General Public License, and you  
welcome to change it and/or distribute copies of it under certain conditi
```

```
Type "show copying" to see the conditions.
```

```
There is absolutely no warranty for GDB. Type "show warranty" for detail
```

```
This GDB was configured as "i386-linux"...Using host libthread_db library
```

```
(gdb)
```

Alguns comandos

help Lista as classes de comandos

```
(gdb) h
List of classes of commands:
[...]
```

run Executa o programa que pode receber um arquivo de dados como entrada:

```
(gdb) r < dados.txt
```

b Coloca um *breakpoint* em uma linha ou função:

```
(gdb) b main
Note: breakpoint 1 also set at pc 0x80483d7.
Breakpoint 2 at 0x80483d7: file trip.c, line
(gdb) b 12
Breakpoint 1 at 0x80483f5: file trip.c, line
```

Mais alguns comandos

next Executa o próximo passo do programa:

```
(gdb) n
10         while (n != 0)
(gdb) n
23         total = 0;
```

until Executa até a linha especificada:

```
(gdb) u 12
```

continue Executa até o próximo *breakpoint* ou até o fim do programa:

```
(gdb) c
Continuing.
```

Mais mais alguns comandos

step Semelhante ao *next* só que “entra” em funções:

```
(gdb) s
24          for (i = 0; i < n; i++)
```

print Mostra o conteúdo de uma variável:

```
(gdb) p i
$6 = 0
```

display Mostra o conteúdo de uma variável a cada passo:

```
(gdb) display i
```

Mais mais mais alguns comandos

watch Para a execução do programa e mostra o conteúdo de uma variável sempre que ela é alterada:

```
(gdb) watch i  
Hardware watchpoint 2: i
```

quit Encerra a execução do gdb:

```
(gdb) quit  
The program is running. Exit anyway? (y or n)
```