

GDB

GNU Debugger

Prof. Kennedy Reurison Lopes

July 30, 2023

Introdução

O **GDB** (GNU Project Debugger) é uma ferramenta para:

- Observar um programa enquanto este executa
- Ver o estado no momento que a execução falha

Permite:

- Iniciar a execução de um programa
- Executar linha-a-linha
- Especificar pontos de paragem
- Imprimir valores de variáveis

Suporta C, C++, Objective-C, Ada e Pascal (entre outras linguagens)

Compilação para depuração

Considere que você possui um arquivo com o código fonte escrito em C com o nome *main.c*.

Compilação Normal:

```
$ gcc -o main main.c
```

Compilação em modo de depuração:

```
$ gcc -o main main.c -g
```

Execução¹

\$ gdb main core dump

Comando para executar um arquivo depurável

\$ gdb --args program args...

Execução de um arquivo depurável com parâmetros.

\$ gdb --pid pid

Executa em modo depuração um programa que já está em execução

¹Cinza → opcional

Execução

set args args...

Modifica os argumentos a serem passados para o programa

run

Roda o programa a ser depurado

kill

Finaliza o programa em depuração

Breakpoints

break where

Cria um novo breakpoint

delete breakpoint#

Remove um breakpoint

enable breakpoint#

Habilita um breakpoint desabilitado

Clear

Apaga todos os breakpoints

Whatchpoints

whatch where

Cria um novo whatchpoint

delete/enable/disable breakpoint#

Semelhante ao breakpoint

Condicionais

break/whatch where if condicao

break/whatch caso determinada condição seja satisfeita

condition breakpoint#

Modifica determinada condição de um break/whatch existente

Examinando a pilha

backtrace/where

Exibe o rastreamento da pilha de execução

backtrace/where full

Exibe o rastreamento da pilha de execução com informações sobre as variáveis locais

frame frame#

Seleciona um determinado frame(um passo na pilha de execução) para operar sobre

Caminhando

step

Caminha para a próxima instrução (linha de código), entrando nas funções

next

Caminha para a próxima instrução (linha de código), **não** entrando nas funções

Caminhando

finish

Continua até que a função corrente retorne

continue

Continua a execução normal

Variáveis e Memória

Comando: `print format var`

Letras que podem ser usadas no format

- o (octal)
- x (hexadecimal)
- d (decimal)
- t (binário)
- f (float)
- a (endereço)
- i (instrução)
- c (caractere de 1 byte)
- s (string)

Variáveis e Memória

display/format what

Mesmo que o print, mas imprime na tela a cada passo(step) de execução

undisplay display#

Remove um display

enable/disable display display#

Remove um display

Código Fonte

list,list filename:function, list filename:line_number

Mostra o código fonte que está sendo depurado

Informações

info args

imprime os argumentos para a função da corrente pilha de execução

info breakpoints

imprime os breakpoints e whatchpoints

info display

imprime os breakpoints e whatchpoints

Informações

info local

imprime as variáveis locais do frame selecionado

whatis variable name

Imprime o tipo da variável