GDB QUICK REFERENCE

Essential Commands

gdb program [core] debug program [using coredump core]
b [file:function] set breakpoint at function [in file]
run [arglist] start your program with [arglist]
bt backtrace: display program stack
p expr display the value of an expression
c continue running your program
n next line, stepping over function calls
s next line, stepping into function calls

Starting GDB

gdb start GDB, with no debugging files

Stopping GDB

quit exit GDB; also q or EOF (eg C-d)

Getting Help

help list classes of commands
help class one-line descriptions for commands in class
help command describe command

Executing your Program

run arglist start your program with arglist
run start your program with current argument list
run <inf>outf start your program with input, output redirected

kill kill running program

tty dev use dev as stdin and stdout for next run
set args arglist specify empty argument list
set args display argument list
show environment show all environment variables
show env var show value of environment variable var
set env var string set environment variable var
unset env var remove var from environment

Shell Commands

cd or change working directory to dr
pwd Print working directory
make ... call `make'
shell cmd execute arbitrary shell command string

Breakpoints and Watchpoints

break [file:line] set breakpoint at line number [in file]
b [file:line] example: break main.c:37
break [file:function] set breakpoint at function [in file]
break +offset set break at offset lines from current stop
break -offset set break at offset lines from current stop
break *addr set breakpoint at address addrreak set breakpoint at next instruction
break ... if expr break conditionally on nonzero expr
cond n [expr] new conditional expression on breakpoint
next line, stepping over function calls
break next line/, stepping into function calls

Execution Control

continue [count] continue running; if count specified, ignore this breakpoint next count times
step [count] execute until another line reached; repeat count times if specified
si [count] step by machine instructions rather than source lines
next [count] execute next line, including any function calls
n [count] next machine instruction rather than source line

Until [location] run until next instruction (or location)
finish run until selected stack frame returns
return [expr] pop selected stack frame without executing [setting return value]
signal num resume execution with signal s (none if 0)
jump line jump line
jump *address evaluate expr without displaying it; use for altering program variables

Program Stack

backtrace [n] print trace of all frames in stack; or of n frames—innermost if n>0, outermost if n<0
bt [n] example: bt
frame [n] select frame number n or frame at address n; if n>0, display current frame
up n select frame n frames up
down n select frame n frames down
info frame [addr] describe selected frame, or frame at addr arguments of selected frame
info locals local variables of selected frame
info reg [rn] ... register values for regs registers in selected frame; all-reg includes floating point
info all-reg [rn] ... information about registers in selected frame
info catch exception handlers active in selected frame

Display

print [f] [expr] show value of expr [or last value] according to format:
   %x hex decimal
   %d signed decimal
   %u unsigned decimal
   %o octal
   %t binary
   %a address, absolute and relative
   %c character
   %f floating point
   % call [f] [expr] like print but does not display void
   %x [NU] expr examine memory at address expr;
     optional format spec follows slash
     count of how many units to display unit size; one of
       b individual bytes
       h halfwords (two bytes)
       w words (four bytes)
       g gigawords (eight bytes)
       p printing format. any printf format, or
       s null-terminated string
       i machine instructions

disassem [addr] display memory as machine instructions

Automatic Display

display [f] [expr] show value of expr each time program
   steps [according to format]
   display
   display all enabled expressions on list
   undisplay n remove number(s) n from list of
   automatically displayed expressions
   disable disp n disable display for expression(s) number n
   enable disp n enable display for expression(s) number n
   info display numbered list of display expressions
Expressions

expr
an expression in C, C++, or Modula-2 (including function calls), or

addr@len an array of len elements beginning at addr

file nm
{type nm
addr
read memory at addr as specified type

$ most recent displayed value

$n nth displayed value as

$n$ nth displayed value back from $n$

$L$ last address examined with $x$

$L$ value at address $L$

$var$ convenience variable; assign any value

show values $n$ show last $n$ values or surrounding $n$

show convenience display all convenience variables

Symbol Table

info address $s$ show where symbol $s$ is stored

info func [reg] show names, types of defined functions (all, or matching reg)

info var [reg] show names, types of global variables (all, or matching reg)

whatis [expr] show data type of expr or $\$ without evaluating; ptype gives more detail

ptype type describe type, struct, union, or enum

GDB Scripts

source script read, execute GDB commands from file script

define cmd create new GDB command cmd execute commandlist

document cmd create online documentation for new GDB command cmd end of documentlist

eend of help-test

Signals

handle signal act specify GDB actions for signal:

print announce signal

noprint be silent for signal

stop halt execution on signal

nostop do not halt execution

pass allow your program to handle signal

nopass do not allow your program to see signal

show table of signals, GDB action for each

Debugging Targets

target type param connect to target machine, process, or file

help target display available targets

attach param connect to another process

detach release target from GDB control

Controlling GDB

set param value set one of GDB’s internal parameters display current setting of parameter

show param

Parameters understood by set and show:

complaints limit number of messages on unusual symbols enable or disable cautionary queries

core limit command-line editing control readline command-line editing

height lpp number of lines before pause in display

language lang Language for GDB expressions [auto, c or modula-2]

listsize n number of lines shown by list

prompt str use str as GDB prompt
defal of, decimal, or hex number representation

radix base control messages when loading symbols

write on/off number of characters before line folded

history ...

groups with the following options:

h view ... disable/enable readline history expansion

h file filename file for recording GDB command history

h size size number of commands kept in history list

save off/on control use of external file for command history

print ...

groups with the following options:

p ...

address on/off print memory addresses in stacks, values

array on/off compact or attractive format for arrays

demangle on/off control use of demangled or internal form for C++ symbols

p asm on/off demangle C++ symbols in machine-instruction output

elements limit number of array elements to display

p object on/off print C++ derived types for objects

p pretty on/off display C++ virtual function tables

p union on/off display union members

p vtbl on/off

p object on/off print C++ derived types for objects

p pretty on/off display C++ virtual function tables

p union on/off display union members

p vtbl on/off display C++ virtual function tables

Source Files

dir names add directory names to front of source path

dir dirnames clear source path

show dir show current source path

list list show next ten lines of source

list - show previous ten lines

list lines display source surrounding lines, specified as:

file num line number in named file

file function beginning of function in named file

toff on/off allow or disable off lines after last printed

address address line containing address

list file list from line $l$ to line $l$

info line num show starting, ending addresses of compiled code for source line num

show sources list all source files in use

show source list all source files in use

show license display GNU General Public License

show warranty there is no warranty for GDB, Display full nowarranty statement.