GDB QUICK REFERENCE  GDB Version 5

Essential Commands
gdb program [core] debug program [using core dump core]
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
gdb program begin debugging program
gdb program core debug core dump core produced by program
gdb --help describe command line options

Stopping GDB
quit exit GDB; also q or EOF (eg C-c)
INTERRUPT (eg C-c) terminate current command, or send to running process

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 arglist for next run
set args specify empty argument list
show args display argument list
show env show value of 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 dir change working directory to dir
pmd Print working directory
make ... call "make"
shell cmd execute arbitrary shell command string

[[ ] surround optional arguments ... show one or more arguments

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 : num
a variable or function num defined in file

{type} addr
read memory at addr as specified type

$ most recent displayed value
$n nth displayed value

$ DISPLAYED previous value to $n
$n nth displayed value back from $n

$ value at address $n

show values [n]
show last 10 values [or surrounding 8n]

Symbol Table

info address s
show where symbol s is stored

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

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

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

tpotype [expr]
describe type, struct, union, or enum

GDB Scripts

source script
read, execute GDB commands from file

define cmd
create new GDB command cmd; execute script defined by command-list

end command-list
end of command-list

document cmd
create online documentation for new GDB command cmd

end end of help-text

Signals

handle signal act
specify GDB actions for signals

print announce signal

nopriment be silent for signal

stop halt execution on signal

nostop do not halt execution

pass allow your program to handle signal

noplus do not allow your program to see signal

info signals 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:

complain limit
number of messages on unusual symbols

can on/off
enable or disable cautionary queries

editing command-line editing

type
number of lines before pause in display

language lang
Language for GDB expressions (auto, c or module-2)

listsize n
number of lines shown by list

prompt str
use str as GDB prompt

radix base
octal, decimal, or hex number representation

verbose on/off
control messages when loading symbols

cin
teraction model before line folded

write on/off
 Allow or forbid patching binary, core files (when recompiled with exec or core)

history
 groups with the following options:

h ... disable/enable readline history expansion

t file filename
file for recording GDB command history

h size size
number of commands kept in history list

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

print ...
 groups with the following options:

 p ... print memory addresses in stacks, values

 p array on/off compact or attractive format for arrays

 p demandl on/off source (demangled) or internal form for C++ symbols

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

 p elements limit number of array elements to display

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

 p pretty on/off struct display; compact or indented

 p union on/off display of union members

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

show commands show last 10 commands

show commands n show 10 commands around number n

show commands + show next 10 commands

Working Files

file [file]
use file for both symbols and executable; with no arg, discard both

core [file]
read file as core dump; or discard

exec [file]
use file as executable only; or discard

symbol [file]
use symbol table from file; or discard

load file
dynamically link file and add its symbols to symbol table

add-sym file addr
dynamically loaded at addr

info files display working files and targets in use

display streams to front of path searched for executable and symbol files

show path list names of shared libraries currently loaded

GDB under GNU Emacs

M-x gdb run GDB under Emacs
C-h m describe GDB mode
C-s step one line (step)
C-n next line (next)
M-1 step one instruction (stepi)
C-c C-f finish current stack frame (finish)
C-c C-c continue (cont)
M-u up args frames (up)
M-d down args frames (down)
C-x & C-x SPAC copy number from point, insert at end

GDB License

show copying Display GNU General Public License
show warranty There is NO WARRANTY for GDB.


The author assumes no responsibility for any errors on this card.

This card may be freely distributed under the terms of the GNU General Public License.

Please contribute to development of this card by annotating it.

Improvements can be sent to bug-gdb@gnu.org.

GDB itself is free software; you are welcome to distribute copies of it under the terms of the GNU General Public License. There is absolutely no warranty for GDB.