GDB QUICK REFERENCE  GDB Version 4

Essential Commands
gdb program core  debug program [using core dump core]
b [file] function set breakpoint at function [in file]
run [arglist] start your program with arglist
print expr display the value of an expression
continue running your program
next line, stepping over function calls
next line, stepping into function calls

Starting GDB
gdb program start GDB, with no debugging files
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 (e.g. C-c)

Getting Help
help list classes of commands
help class list descriptions of commands in class
help command describe command

Executing Your Program
run [arglist] start your program with arglist
run...infout start your program with input, output redirected
run...inf out start your program with input, output redirected

kill kill running program

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

Shell Commands
cd dir change working directory to dir
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 break at line in file
break [file] function set breakpoint at function [in file]
break + offset set break at offset lines from current break
break - offset break conditionally on non-zero expr
break... if expr break conditionally on non-zero expr
cond n [expr] new conditional expression on breakpoint n; make unconditional if no expr
tbreak... temporary break; disable when reached
break regex break on all functions matching regex
watch expr set a watchpoint for expression
break at C++ handler for exception z
info break show defined breakpoints
info watch show defined watchpoints
clear delete breakpoints at next interruption
clear [file] function delete breakpoints at entry to function
break [file] line delete breakpoints on source line
delete n delete breakpoints or breakpoint n
disable n disable breakpoints or breakpoint n
enable n enable breakpoints or breakpoint n
enable once n disable again when reached
enable dal n disable breakpoints or breakpoint n; disable when reached
ignore n count ignore breakpoint n, count times
commands n count execute GDB commands n every time breakpoint n is reached
silent commands list suppress default display
end end of commands list

Program Stack
backtrace print trace of all frames in stack; or of n frames: innermost if n>0, outermost if n<0
bt n print frame number n or frame at address n; if no frame, 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
info args arguments of selected frame
info locals local variables of selected frame
info reg [addr] register values (orregs) in selected frame; all-reg includes floating point
info catch exception handlers active in selected frame

Execution Control
continue [count] continue running; if count specified, ignore this breakpoint next count times
stop [count] execute until another line reached; repeat count times if specified
step [count] step by machine instructions rather than source lines
next [count] execute next line, including any function calls
nexti [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 [more if 0]
jump line execute at specified line number or address
jump address evaluate expr without displaying it; use for altering program variables

Display
print /f] [expr] show value of expr [or last value]$ according to format
p /f] [expr] hexadecimal
x d signed decimal
u u unsigned decimal
o octal
b binary
a address, absolute and relative
c character
f floating point

N call /f] [expr] like print but does not display void
x/Nuf] expr examine memory at address expr; optional format spec follows slash
N count of how many units to display
u unit spec, one of
b individual bytes
h halfwords (two bytes)
w words (four bytes)
g great words (eight bytes)
/ printing format. Any print format, or
f machine instructions

Disassemble [addr] display memory as machine instructions

Automatic Display
display /f] [expr] show value of expr each time program steps according to format
break 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) n
enable disp n enable display for expression(s) n
info display numbered list of display expressions
Expressions

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

a variable or function nm defined in file

(file: nm)
read memory at addr as specified type

(type) addr
most recent displayed value

n
nth displayed value

$$(n)
value at address $n$

$var
convenience variable; assign any value

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

show conv
display all convenience variables

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)

what in [expr]
show data type of expr [or $]
without evaluating; ptype gives more details

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

GDB Scripts

script
read, execute GDB commands from file

define cmd
create new GDB command cmd; execute
script defined by commandlist

define list
end commandlist

document cmd
create online documentation for new GDB
command cmd

document list
end of helpertext

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

nops
allow your program to use signal

info signals
show table of signals, GDB action for each

Debugging Targets

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

display available targets

help target
connect to another process

attach param
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:

complaint
print number of messages on unusual symbols

confirm on/off
enable or disable customary queries

editing on/off
control readline commandline editing

height by
number of lines before pause in display

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

lists size n
number of lines shown by list

prompt str
use str as GDB prompt

radix base
output, decimal, or hex number

representation

verbos e on/off
control messages when loading symbols

width opt
number of characters before line folded

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

history...

h ...
disable/enable readline history expansion

h file filename
file for recording GDB command history

h size size
number of commands kept in history list

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

print...
groups with the following options:

p ...

p address on/off
print memory addresses in stack, values

p array off/on
compact or attractive format for arrays

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

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

p elements
limit number of heap elements to display

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

p pretty off/on
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 command list, numbered

show commands n
show 10 commands around number n

show commands +
show next 10 commands

Source Files

list
to current source path

list dir
to current source path

list -
display source line numbers, specified as:

[re]num
line number [in named file]

[re]func
beginning of function [in named file]

toff
off lines after last printed

avoid
off lines previous to last printed

address
line containing address

list f, f to line l
show starting, ending addresses of
compiled code for source line num

info source
show names of current source file

list all source files in use

info sources
list all source files in use

forw regex
search following source lines for regex

rew regex
search preceding source lines for regex

GDB under GNU Emacs

Max
run GDB under Emacs

C-h m
describe GDB mode

C-h s
step one line [step]

C-h i
step one instruction [step]

C-c C-f
finish current stack frame [finish]

C-c cont
continue [cont]

C-u up
up arg frames [up]

C-d
down arg frames [down]

C-g &
copy number from point, insert at end

C-x S/C
(in source file) set break at point

GDB License

show copying
Display GNU General Public License

show warranty
There is NO WARRANTY for GDB.

Display full nonguarantee statement.


The author assumes no responsibility for any errors in 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.

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