## GDB QUICK REFERENCE GDB Version 5

#### **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 backtrace: display program stack bt display the value of an expression p expr continue running your program next line, stepping over function calls next line, stepping into function calls

### Starting GDB

gdb start GDB, with no debugging files gdb program begin debugging program gdb program core debug coredump core produced by

gdb --help describe command line options

## Stopping GDB

quit exit GDB; also q or EOF (eg C-d) 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 aralist start your program with arglist

riin start your program with current argument

run ... <inf >outf start your program with input, output

redirected

kill kill running program

tty devuse dev as stdin and stdout for next run

set args arglist specify aralist for next run specify empty argument list set args

show args display argument list

show env show all environment variables

show env var show value of environment variable var

set environment variable var set env var string unset env var remove var from environment

## Shell Commands

cd dirchange working directory to dir

bwd Print working directory

make ... call "make"

shell cmd execute arbitrary shell command string

surround optional arguments ... show one or more arguments

#### (c)1998-2018 Free Software Foundation, Inc. Permissions on back

## **Breakpoints and Watchpoints**

break [file:]line set breakpoint at line number in file b [file:]line eg: break main.c:37 break [file:]func set breakpoint at func in file break +offset set break at offset lines from current stop break -offset break \* addrset breakpoint at address addrbreak set breakpoint at next instruction break ... if exprbreak conditionally on nonzero expr cond n |expr|new conditional expression on breakpoint n; make unconditional if no expr tbreak ... temporary break; disable when reached rbreak [file: regex break on all functions matching regex in file watch exprset a watchpoint for expression expr break at event, which may be catch,

catch event throw, exec, fork, vfork, load, or

unload.

info break show defined breakpoints info watch show defined watchpoints

clear delete breakpoints at next instruction clear | file: | fun delete breakpoints at entry to fun() clear [file: line delete breakpoints on source line delete [n]delete breakpoints or breakpoint n

disable [n]disable breakpoints or breakpoint nenable [n]enable breakpoints or breakpoint nenable once |n|enable breakpoints or breakpoint n; disable again when reached

enable del |n|enable breakpoints or breakpoint n; delete when reached

ignore n count ignore breakpoint n, count times

execute GDB command-list every time commands nsilent breakpoint n is reached. silent command-list suppresses default display

end of command-list end

## Program Stack

backtrace [n]print trace of all frames in stack; or of nframes—innermost if n>0, outermost if bt [n]n < 0frame nselect frame number n or frame at address n; if no n, display current frame select frame n frames up up n ${\tt down}\ n$ 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 [rn]... register values for regs rn in selected frame; all-reg includes floating point info all-reg [rn]

## **Execution Control**

$\begin{array}{c} \texttt{continue} \ \left[ count \right] \\ \texttt{c} \ \left[ count \right] \end{array}$	continue running; if <i>count</i> specified, ignore this breakpoint next <i>count</i> times
$\begin{array}{l} \mathtt{step} \ \big[ count \big] \\ \mathtt{s} \ \big[ count \big] \end{array}$	execute until another line reached; repeat $count\ {\rm times}$ if specified
$\begin{array}{l} \mathtt{stepi} \ \big[ count \big] \\ \mathtt{si} \ \big[ count \big] \end{array}$	step by machine instructions rather than source lines $% \left( 1\right) =\left( 1\right) \left( 1\right)$
$\begin{array}{l} \texttt{next} \ \big[ count \big] \\ \texttt{n} \ \big[ count \big] \end{array}$	execute next line, including any function calls
$\begin{array}{l} {\tt nexti} \ \big[ count \big] \\ {\tt ni} \ \big[ count \big] \end{array}$	next machine instruction rather than source line
${ t until} \ igl[ location igr]$	run until next instruction (or location)
finish	run until selected stack frame returns
$\texttt{return} \ \left[ expr \right]$	pop selected stack frame without executing [setting return value]
$\verb signal  num $	resume execution with signal $s$ (none if $0$ )
jump line	resume execution at specified <i>line</i> number or address
jump *address	
set var=expr	evaluate <i>expr</i> without displaying it; use for altering program variables

## Display

Display	
$\begin{array}{c} \mathtt{print} \ \left[ / f \right] \left[ expr \right] \\ \mathtt{p} \ \left[ / f \right] \left[ expr \right] \end{array}$	show value of $expr$ [or last value $\$$ ] according to format $f$ :
P[I][expI]	
x	hexadecimal
d	signed decimal
u	unsigned decimal
0	octal
t	binary
a	address, absolute and relative
С	character
f	floating point
$\mathtt{call} \ ig[/fig] \ expr$	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 size; one of
	b individual bytes
	h halfwords (two bytes)
	w words (four bytes)
	g giant words (eight bytes)
f	printing format. Any print format, or
	s null-terminated string
	i machine instructions
${\tt disassem} \; \big[ addr \big]$	display memory as machine instructions

## Automatic Display

$\mathtt{display} \; \big[ / f \big] \; expr$	show value of $expr$ each time program stops [according to format $f$ ]		
display	display all enabled expressions on list		
$\verb"undisplay" n$	remove number(s) $n$ from list of automatically displayed expressions		
$\hbox{\tt disable disp } n$	disable display for expression(s) number $n$		
enable disp $\it n$ info display	enable display for expression(s) number $n$ 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	a variable or function $nm$ defined in $file$
$\{type\}addr$	read memory at $addr$ as specified $type$
\$	most recent displayed value
\$n	nth displayed value
\$\$	displayed value previous to \$
n	nth displayed value back from \$
\$_	last address examined with x
\$	value at address \$_
var	convenience variable; assign any value
show values $ig[nig]$	show last 10 values [or surrounding $n$ ]

display all convenience variables

## Symbol Table

show conv

${ t info}$ address $s$	show where symbol $s$ is stored
$\verb info func  [regex] $	show names, types of defined functions (all, or matching regex)
$\verb"info var" \left[ \textit{regex} \right]$	show names, types of global variables (all, or matching $regex$ )
whatis $\begin{bmatrix} expr \end{bmatrix}$ ptype $\begin{bmatrix} expr \end{bmatrix}$	show data type of $expr$ [or \$] without evaluating; ptype gives more detail
ptype type	describe type, struct, union, or enum

whatis $[expr]$ ptype $[expr]$ ptype $type$	show data type of expr [or \$] without evaluating; ptype gives more detail describe type, struct, union, or enum
GDB Scripts	
source $script$	read, execute GDB commands from file $script$
$\begin{array}{c} \texttt{define} \ cmd \\ command\text{-}list \end{array}$	create new GDB command cmd; execute script defined by command-list
end	end of $command$ -list
$\begin{array}{c} \texttt{document} \ \ cmd \\ help\text{-}text \\ \texttt{end} \end{array}$	create online documentation for new GDB command cmd end of help-text
ciid	end of neip-text

## **Signals**

${ t handle} \ signal \ act$	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
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 show param  Parameters understood by set and show:  complaint limit confirm on/off editing on/off height lpp language lang language lang radix base on/off width cpl write on/off history h exp off/on h file filename h size size h save off/on p armay off/on p demangl on/off print memory addresses in stacks, values p array off/on p union on/off p rottle following of the pression of the following of the pression on/off print C++ derived types for objects show commands show commands show last 10 commands show set	Controlling GDB			
Parameters understood by set and show:  complaint limit  confirm on/off editing on/off height lpp language lang language lang lasse  verbose on/off width cpl write on/off history h exp off/on h file filename h size size h save off/on p address on/off parameters understood by set and show:  complaint limit number of messages on unusual symbols control readline command-line editing number of lines before pause in display Language for GDB expressions (auto, c or modula-2) listsize n number of lines shown by list use str as GDB prompt octal, decimal, or hex number representation control messages when loading symbols number of characters before line folded Allow or forbid patching binary, core files (when reopened with exec or core) groups with the following options: h h exp off/on history  print groups with the following options: p p address on/off print memory addresses in stacks, values p array off/on p demangl 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 p pretty off/on p union on/off display of union members display of union members display of c++ virtual function tables show commands	$\mathtt{set}\ param\ value$	set one of GDB's internal parameters		
complaint limit confirm on/off editing on/off height lpp language lang language lang listsize n prompt str radix base verbose on/off width cpl write on/off h size size h save off/on p address on/off p address on/off p address on/off p address on/off p asm—dem on/off p asm—dem on/off p p retty off/on p union on/off p vtbl off/on sheight lpp language lang language for GDB expressions (auto, c or modula-2) number of lines shown by list use str as GDB prompt octal, decimal, or hex number representation control messages when loading symbols number of characters before line folded Allow or forbid patching binary, core files (when reopened with exec or core) groups with the following options: h h exp off/on h file filename h size size h save off/or p address on/off print memory addresses in stacks, values p array off/or p pretty off/on p union on/off demangle C++ symbols in machine- instruction output p elements limit number of array elements to display p object on/off p vtbl off/on show commands show last 10 commands	show param	display current setting of parameter		
complaint limit confirm on/off editing on/off height lpp language lang language lang listsize n prompt str radix base verbose on/off width cpl write on/off h size size h save off/on p address on/off p address on/off p address on/off p address on/off p asm—dem on/off p asm—dem on/off p p retty off/on p union on/off p vtbl off/on sheight lpp language lang language for GDB expressions (auto, c or modula-2) number of lines shown by list use str as GDB prompt octal, decimal, or hex number representation control messages when loading symbols number of characters before line folded Allow or forbid patching binary, core files (when reopened with exec or core) groups with the following options: h h exp off/on h file filename h size size h save off/or p address on/off print memory addresses in stacks, values p array off/or p pretty off/on p union on/off demangle C++ symbols in machine- instruction output p elements limit number of array elements to display p object on/off p vtbl off/on show commands show last 10 commands				
editing on/off height lpp language lang language lang listsize n prompt str radix base verbose on/off width cpl write on/off history h exp off/on h size size h save off/on print print print paddress on/off print memory addresses in stacks, values p array off/on p asm-dem on/off p lements limit p elements limit p other of onlines before pause in display language for GDB expressions (auto, c or modula-2) number of lines shown by list use str as GDB prompt octal, decimal, or hex number representation control messages when loading symbols number of characters before line folded Allow or forbid patching binary, core files (when reopened with exec or core) groups with the following options: h disable/enable readline history expansion file for recording GDB command history number of commands kept in history list control use of external file for command history  print groups with the following options: p p address on/off print memory addresses in stacks, values p array off/on control use of external file for command history  print p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl on/off demangle C++ symbols in machine- instruction output p elements limit number of array elements to display p object on/off p print C++ derived types for objects struct display: compact or indented display of union members display of C++ virtual function tables show commands show last 10 commands				
editing on/off height lpp language lang language lang listsize n prompt str radix base verbose on/off width cpl write on/off history h exp off/on h size size h save off/on print print print paddress on/off print memory addresses in stacks, values p array off/on p asm-dem on/off p lements limit p elements limit p other of onlines before pause in display language for GDB expressions (auto, c or modula-2) number of lines shown by list use str as GDB prompt octal, decimal, or hex number representation control messages when loading symbols number of characters before line folded Allow or forbid patching binary, core files (when reopened with exec or core) groups with the following options: h disable/enable readline history expansion file for recording GDB command history number of commands kept in history list control use of external file for command history  print groups with the following options: p p address on/off print memory addresses in stacks, values p array off/on control use of external file for command history  print p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl on/off demangle C++ symbols in machine- instruction output p elements limit number of array elements to display p object on/off p print C++ derived types for objects struct display: compact or indented display of union members display of C++ virtual function tables show commands show last 10 commands	$confirm \ on/off$	enable or disable cautionary queries		
language lang Language for GDB expressions (auto, c or modula-2)  listsize n number of lines shown by list use str as GDB prompt octal, decimal, or hex number representation verbose on/off width cpl number of characters before line folded write on/off Allow or forbid patching binary, core files (when reopened with exec or core) history groups with the following options: h h exp off/on disable/enable readline history expansion h file filename file for recording GDB command history number of commands kept in history list control use of external file for command history  print groups with the following options: p p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl 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 off/on struct display: compact or indented display of union members p vtbl off/on display of union members show commands show last 10 commands	,			
listsize n number of lines shown by list prompt str use str as GDB prompt octal, decimal, or hex number representation verbose on/off control messages when loading symbols midth cpl number of characters before line folded write on/off Allow or forbid patching binary, core files (when reopened with exec or core) history groups with the following options: h h exp off/on disable/enable readline history expansion h file filename file for recording GDB command history number of commands kept in history list control use of external file for command history  print groups with the following options: p p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl 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 off/on struct display: compact or indented display of union members p vtbl off/on display of C++ virtual function tables show commands show last 10 commands	$\texttt{height}\ lpp$	number of lines before pause in display		
prompt str radix base  octal, decimal, or hex number representation  control messages when loading symbols number of characters before line folded Allow or forbid patching binary, core files (when reopened with exec or core) history hexp off/on h file filename h size size h save off/on  print p address on/off print memory addresses in stacks, values p array off/on p demangl on/off demangle C++ symbols p asm-dem on/off p print C++ derived types for objects p vtbl off/on  sizel size use str as GDB prompt octal, decimal, or hex number representation control messages when loading symbols number of characters before line folded Allow or forbid patching bymany, core files (when reopened with exec or core) groups with the following options: h p disable/enable readline history expansion file for recording GDB command history number of commands kept in history list control use of external file for command history  groups with the following options: p p address on/off print memory addresses in stacks, values compact or attractive format for arrays p demangl on/off 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 p print C++ derived types for objects struct display: compact or indented display of union members display of C++ virtual function tables	${\tt language}\ lang$			
radix base  octal, decimal, or hex number representation  verbose on/off width cpl control messages when loading symbols number of characters before line folded allow or forbid patching binary, core files (when reopened with exec or core)  history groups with the following options:  h disable/enable readline history expansion file for recording GDB command history number of commands kept in history list control use of external file for command history  print groups with the following options:  p of command history  p demangle for command history  p demangle for commands 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 off/on display of union members  display of C++ virtual function tables  show commands	listsize $n$	number of lines shown by list		
representation  verbose on/off width cpl write on/off Allow or forbid patching binary, core files (when reopened with exec or core) history h h exp off/on h file filename h size size h save off/on  print  p address on/off print memory addresses in stacks, values p array off/on p asm-dem on/off p elements limit p elements limit p union on/off p vtbl off/on p vtbl off/on  vorted the representation control messages when loading symbols number of characters before line folded Allow or forbid patching binary, core files (when reopened with exec or core) groups with the following options: number of commands kept in history list control use of external file for command history print groups with the following options: p p address on/off print memory addresses in stacks, values compact or attractive format for arrays p demangle C++ symbols in machine- instruction output p elements limit number of array elements to display p object on/off p vtbl off/on display of union members display of union members display of C++ virtual function tables show commands	${ t prompt} \ str$	use $str$ as GDB prompt		
width cpl write on/off Allow or forbid patching binary, core files (when reopened with exec or core) history hexp off/on h file filename h size size h save off/on  print groups with the following options: h control use of external file for command history  print groups with the following options: h control use of external file for command history  print p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl 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 p union on/off display of union members p vtbl off/on display of C++ virtual function tables show commands show last 10 commands	${ t radix}\ base$			
write on/off Allow or forbid patching binary, core files (when reopened with exec or core) history groups with the following options: h h exp off/on disable/enable readline history expansion h file filename file for recording GDB command history number of commands kept in history list control use of external file for command history  print groups with the following options: p p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl 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 off/on struct display: compact or indented display of union members p vtbl off/on display of C++ virtual function tables show commands show last 10 commands	verbose on/off	control messages when loading symbols		
(when reopened with exec or core) history h exp off/on h file filename h size size h save off/on  print paddress on/off print memory addresses in stacks, values parray off/on compact or attractive format for arrays p demangl 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 p print C++ derived types for objects p vtbl off/on show commands show last 10 commands	$\verb width   cpl $	number of characters before line folded		
h h exp off/on h file filename h size size number of commands kept in history list control use of external file for command history  print p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl 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 p print C++ derived types for objects p pretty off/on p union on/off display of union members display of C++ virtual function tables show commands show last 10 commands	write $on/off$			
h exp off/on h file filename h size size h save off/on  print  p address on/off print memory addresses in stacks, values p array off/on p demangl 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 p print C++ derived types for objects p vtbl off/on disable/enable readline history expansion file for recording GDB command history number of external file for command history groups with the following options: p compact or attractive format for arrays p demangl 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 struct display: compact or indented display of union members display of C++ virtual function tables show commands	history	groups with the following options:		
h file filename h size size h save off/on  print  p address on/off print memory addresses in stacks, values p array off/on  p demangl 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 p print C++ derived types for objects p pretty off/on p union on/off display of union members p vtbl off/on  show commands  show last 10 commands				
h size size h save off/on  number of commands kept in history list control use of external file for command history  print  p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl 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 off/on p union on/off display of union members p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands				
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 stacks, values p array off/on compact or attractive format for arrays p demangl on/off source (demangled) or internal form for	•	· ·		
print groups with the following options:  p  p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl on/off source (demangled) or internal form for		* *		
p  p address on/off print memory addresses in stacks, values  p array off/on compact or attractive format for arrays  p demangl 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 off/on struct display: compact or indented  p union on/off display of union members  p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands	n save off/on			
p address on/off print memory addresses in stacks, values p array off/on compact or attractive format for arrays p demangl on/off source (demangled) or internal form for	print	groups with the following options:		
p array off/on compact or attractive format for arrays p demangl 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 off/on struct display: compact or indented p union on/off display of union members p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands	•			
p demangl 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 off/on struct display: compact or indented  p union on/off display of union members  p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands	p address $on/off$	f print memory addresses in stacks, values		
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 off/on struct display: compact or indented  p union on/off display of union members  p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands	${ t p}$ array $o\!f\!f\!/on$	compact or attractive format for arrays		
instruction output  p elements limit number of array 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 off/on display of C++ virtual function tables  show commands show last 10 commands	p demangl $on/of$	,		
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 off/on display of C++ virtual function tables  show commands show last 10 commands	p asm-dem $on/off$			
p pretty off/on struct display: compact or indented punion on/off display of union members p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands	p elements $limit$	number of array elements to display		
p pretty off/on struct display: compact or indented punion on/off display of union members p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands	p object $on/off$	print C++ derived types for objects		
p union on/off display of union members p vtbl off/on display of C++ virtual function tables  show commands show last 10 commands	p pretty off/on	struct display: compact or indented		
show commands show last 10 commands				
	p vtbl off/on	display of C++ virtual function tables		
	show commands	show last 10 commands		
	$\verb show  commands  n$	show 10 commands around number $n$		

# show commands + Working Files

working rifes	
$\mathtt{file} \; \big[ \mathit{file} \big]$	use file for both symbols and executable; with no arg, discard both
$\verb"core" \left[ \mathit{file} \right]$	read file as coredump; or discard
$\verb"exec" [file]"$	use $file$ as executable only; or discard
${\tt symbol} \ \big[ file \big]$	use symbol table from file; or discard
load file	dynamically link file and add its symbols
${\tt add-sym}\ file\ addr$	read additional symbols from file,
-	dynamically loaded at $addr$
info files	display working files and targets in use
path dirs	add dirs to front of path searched for
	executable and symbol files
show path	display executable and symbol file path
info share	list names of shared libraries currently

loaded

show next 10 commands

#### Source Files

dir names

dir

ters understood by <b>set</b> and <b>show</b> :		ull	cicar boarce path
${ t laint} \ limit$	number of messages on unusual symbols	show dir	show current source path
$egin{array}{ll} { m irm} & on/off \ { m ing} & on/off \ { m ht} & lpp \ { m uage} & lang \end{array}$	enable or disable cautionary queries control <b>readline</b> command-line editing number of lines before pause in display Language for GDB expressions (auto, c or	list list - list lines	show next ten lines of source show previous ten lines display source surrounding <i>lines</i> , specified as:
${ t size}  n$ ${ t pt}  str$	modula-2) number of lines shown by list use $str$ as GDB prompt	$egin{aligned} [\mathit{file:}] num \ &[\mathit{file:}] function \end{aligned}$	line number [in named file] beginning of function [in named file]
x base	octal, decimal, or hex number representation	+ off - off	off lines after last printed off lines previous to last printed
ose $on/off$	control messages when loading symbols	*address	line containing address
h $cpl$	number of characters before line folded	$\mathtt{list}\ f, l$	from line $f$ to line $l$
e on/off	Allow or forbid patching binary, core files (when reopened with exec or core)	$\verb info  line  num $	show starting, ending addresses of compiled code for source line <i>num</i>
ory	groups with the following options:	info source	show name of current source file
•		info sources	list all source files in use
off/on	disable/enable readline history expansion	${ t forw}\ regex$	search following source lines for regex
le filename	file for recording GDB command history	rev regex	search preceding source lines for regex

## GDB under GNU Emacs

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

add directory names to front of source

clear source path

## **GDB** License

show copying	Display GNU General Public License
show warranty	There is NO WARRANTY for GDB.
	Display full no-warranty statement.

Copyright © 1991-2018 Free Software Foundation, Inc. Author: Roland H. Pesch

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.