Instantly share code, notes, and snippets.

rkubik / **cheat_sheet.txt**

Last active Nov 19, 2020

☆ Star

`<>` **Code**   -o- Revisions 3   ☆ Stars 128   ⑂ Forks 53

GDB cheat sheet

`<>` **cheat_sheet.txt**

```
 1    GDB commands by function - simple guide
 2    --------------------------------------
 3    More important commands have a (*) by them.
 4
 5    Startup
 6    % gdb -help          print startup help, show switches
 7    *% gdb object         normal debug
 8    *% gdb object core    core debug (must specify core file)
 9    %% gdb object pid     attach to running process
10    % gdb                use file command to load object
11
12    Help
13    *(gdb) help           list command classes
14    (gdb) help running    list commands in one command class
15    (gdb) help run        bottom-level help for a command "run"
16    (gdb) help info       list info commands (running program state)
17    (gdb) help info line  help for a particular info command
18    (gdb) help show       list show commands (gdb state)
19    (gdb) help show commands      specific help for a show command
20
21    Breakpoints
22    *(gdb) break main     set a breakpoint on a function
23    *(gdb) break 101      set a breakpoint on a line number
24    *(gdb) break basic.c:101      set breakpoint at file and line (or function)
25    *(gdb) info breakpoints        show breakpoints
26    *(gdb) delete 1       delete a breakpoint by number
27    (gdb) delete          delete all breakpoints (prompted)
28    (gdb) clear           delete breakpoints at current line
29    (gdb) clear function  delete breakpoints at function
30    (gdb) clear line      delete breakpoints at line
31    (gdb) disable 2       turn a breakpoint off, but don't remove it
32    (gdb) enable 2        turn disabled breakpoint back on
```

```
33   (gdb) tbreak function|line        set a temporary breakpoint
34   (gdb) commands break-no ... end   set gdb commands with breakpoint
35   (gdb) ignore break-no count       ignore bpt N-1 times before activation
36   (gdb) condition break-no expression       break only if condition is true
37   (gdb) condition 2 i == 20         example: break on breakpoint 2 if i equals 20
38   (gdb) watch expression        set software watchpoint on variable
39   (gdb) info watchpoints        show current watchpoints
40
41   Running the program
42   *(gdb) run                run the program with current arguments
43   *(gdb) run args redirection       run with args and redirection
44   (gdb) set args args...        set arguments for run
45   (gdb) show args       show current arguments to run
46   *(gdb) cont               continue the program
47   *(gdb) step               single step the program; step into functions
48   (gdb) step count      singlestep \fIcount\fR times
49   *(gdb) next               step but step over functions
50   (gdb) next count      next \fIcount\fR times
51   *(gdb) CTRL-C         actually SIGINT, stop execution of current program
52   *(gdb) attach process-id      attach to running program
53   *(gdb) detach        detach from running program
54   *(gdb) finish        finish current function's execution
55   (gdb) kill           kill current executing program
56
57   Stack backtrace
58   *(gdb) bt                print stack backtrace
59   (gdb) frame              show current execution position
60   (gdb) up                 move up stack trace  (towards main)
61   (gdb) down               move down stack trace (away from main)
62   *(gdb) info locals     print automatic variables in frame
63   (gdb) info args        print function parameters
64
65   Browsing source
66   *(gdb) list 101        list 10 lines around line 101
67   *(gdb) list 1,10       list lines 1 to 10
68   *(gdb) list main       list lines around function
69   *(gdb) list basic.c:main       list from another file basic.c
70   *(gdb) list -          list previous 10 lines
71   (gdb) list *0x22e4     list source at address
72   (gdb) cd dir           change current directory to \fIdir\fR
73   (gdb) pwd              print working directory
74   (gdb) search regexpr    forward current for regular expression
75   (gdb) reverse-search regexpr       backward search for regular expression
76   (gdb) dir dirname      add directory to source path
77   (gdb) dir              reset source path to nothing
78   (gdb) show directories       show source path
79
80   Browsing Data
81   *(gdb) print expression       print expression, added to value history
```

```
 82   *(gdb) print/x expressionR        print in hex
 83   (gdb) print array[i]@count         artificial array - print array range
 84   (gdb) print $           print last value
 85   (gdb) print *$->next    print thru list
 86   (gdb) print $1          print value 1 from value history
 87   (gdb) print ::gx        force scope to be global
 88   (gdb) print 'basic.c'::gx        global scope in named file (>=4.6)
 89   (gdb) print/x &main     print address of function
 90   (gdb) x/countFormatSize address      low-level examine command
 91   (gdb) x/x &gx           print gx in hex
 92   (gdb) x/4wx &main       print 4 longs at start of \fImain\fR in hex
 93   (gdb) x/gf &gd1         print double
 94   (gdb) help x           show formats for x
 95   *(gdb) info locals     print local automatics only
 96   (gdb) info functions regexp        print function names
 97   (gdb) info variables  regexp       print global variable names
 98   *(gdb) ptype name        print type definition
 99   (gdb) whatis expression      print type of expression
100   *(gdb) set variable = expression        assign value
101   (gdb) display expression      display expression result at stop
102   (gdb) undisplay        delete displays
103   (gdb) info display     show displays
104   (gdb) show values      print value history (>= gdb 4.0)
105   (gdb) info history     print value history (gdb 3.5)
106
107   Object File manipulation
108   (gdb) file object              load new file for debug (sym+exec)
109   (gdb) file                     discard sym+exec file info
110   (gdb) symbol-file object       load only symbol table
111   (gdb) exec-file object         specify object to run (not sym-file)
112   (gdb) core-file core           post-mortem debugging
113
114   Signal Control
115   (gdb) info signals             print signal setup
116   (gdb) handle signo actions     set debugger actions for signal
117   (gdb) handle INT print         print message when signal occurs
118   (gdb) handle INT noprint       don't print message
119   (gdb) handle INT stop          stop program when signal occurs
120   (gdb) handle INT nostop        don't stop program
121   (gdb) handle INT pass          allow program to receive signal
122   (gdb) handle INT nopass        debugger catches signal; program doesn't
123   (gdb) signal signo             continue and send signal to program
124   (gdb) signal 0                 continue and send no signal to program
125
126   Machine-level Debug
127   (gdb) info registers           print registers sans floats
128   (gdb) info all-registers       print all registers
129   (gdb) print/x $pc              print one register
130   (gdb) stepi                    single step at machine level
```

```
131   (gdb) si                       single step at machine level
132   (gdb) nexti                    single step (over functions) at machine level
133   (gdb) ni                       single step (over functions) at machine level
134   (gdb) display/i $pc            print current instruction in display
135   (gdb) x/x &gx                  print variable gx in hex
136   (gdb) info line 22             print addresses for object code for line 22
137   (gdb) info line *0x2c4e        print line number of object code at address
138   (gdb) x/10i main               disassemble first 10 instructions in \fImain\fR
139   (gdb) disassemble addr         dissassemble code for function around addr
140
141   History Display
142   (gdb) show commands            print command history (>= gdb 4.0)
143   (gdb) info editing             print command history (gdb 3.5)
144   (gdb) ESC-CTRL-J               switch to vi edit mode from emacs edit mode
145   (gdb) set history expansion on     turn on c-shell like history
146   (gdb) break class::member      set breakpoint on class member. may get menu
147   (gdb) list class::member       list member in class
148   (gdb) ptype class              print class members
149   (gdb) print *this              print contents of this pointer
150   (gdb) rbreak regexpr           useful for breakpoint on overloaded member name
151
152   Miscellaneous
153   (gdb) define command ... end      define user command
154   *(gdb) RETURN                  repeat last command
155   *(gdb) shell command args      execute shell command
156   *(gdb) source file             load gdb commands from file
157   *(gdb) quit                    quit gdb
```