GDB

```
set disassembly-flavor intel

$ cat ~/.bash_aliases | grep gdb
alias gdb='gdb -quiet'

Running gdb
------------------
$ gdb        ■■- run, then use file command to load object
$ gdb -quiet        - supress copyright information
$ gdb object      ■- normal debug
$ gdb object core ■- analyze core dump
$ gdb object pid  ■- attach to running process

General commands
------------------
set args                         - set program arguments
show args                        - show program arguments
run                              - run the program
run < file                       - run with input from file
set follow-exec-mode new/sam     - set debugger response to an exec call
set write                        - set write into executables
set write off                    - unset write int oexecutables
continue                         - continue running until break
finish                           - execute until current stack frame ends
source FILE                      - read commands from script file
shell  [cmd]                     - run cmd in a shell
display /5i $eip                 - display expression everytime execution stops
undisplay <expr #>               - undisplay expression number
info functions                   - list all the functions
info variables                   - list all the variables
info registers                   - list most common registers
info all-registers               - list all registers
info display                     - print the list of displayed expressions
backtrace                        - print backtrace of all stack frames
where                            - same as backtrace
set disassembly-flavor intel     - set disassembly style to intel/att
define hook-[cmd]                - actions to execute before command
define hooopost-[cmd]            - actions to execute after command
define hook-stop                 - actions to execute when execution stops

Breakpoints
------------------
info breakpoints                 - list all breakpoints
break [func]                     - break function name
break *[addr]                    - break at address
delete [bnum]                    - delete breakpoint bnum
break if [cond]                  - break if condition
ignore [bnum] [count]            - ignore breakpoint bnum count times
condition [bnum] $eax == 0x22    - add condition for breakpoint 1
condition [bnum]                 - delete condition for breakpoint 1

Watchpoints
------------------
info watchpoints                 - list all the watchpoint
watch variable==value            - break when variable equals ..
watch $eax == 0x0000ffaa         - break when register equals ..
rwatch *[addr]                   - break on read memory location
awatch *[addr]                   - break on read/write memory location
```