Starting:
gdb
gdb <file>

Run and stop:
help Get information about gdb
quit Exit gdb
run Run program
run 1 2 3 Run program with command-line arguments 1 2 3
run < in.txt Run program with input redirected from in.txt
kill Stop the program
Control-D Exit gdb
Control-C Stop the currently running gdb command
(Does not exit GDB.)
make Run make to rebuild without leaving gdb

Breakpoints:
b break sum Set breakpoint at the entry to function sum
b break *0x80483c3 Set breakpoint at address 0x80483c3
(review gdb numbers each breakpoint you create)
delete 1 Delete breakpoint 1
disable 1 Disable breakpoint 1
enable 1 Enable breakpoint 1
delete Delete all breakpoints
clear sum Clear breakpoints at entry to function sum

Examine Code:
disas Disassemble current function
disas sum Disassemble function around 0x80483b7
disas 0x80483b7 Disassemble code within specified address range
print /x $eip Print program counter in hex
print /d $eip Print program counter in decimal
print /t $eip Print program counter in binary

Examine Data:
print /d $eax Print contents of $eax in decimal
print /t $eax Print contents of $eax in hex
print /x $eax Print contents of $eax in binary
print 0x100 Print decimal representation of 0x100
print /x 555 Print hex representation of 555
print /x ($esp+8) Print (contents of $esp) + 8 in hex
print *(int *) 0xbffff890 Print integer at address 0xbffff890
print *(char *) 0xbffff890 Print string at address 0xbffff890
x/w 0xbffff890 Examine 4-byte word starting at address 0xbffff890
x/wd $esp Examine 4-byte word starting at address in $esp
x/wd $esp Examine 4-byte word starting at address in $esp
    in decimal
x/2w $esp Examine two 4-byte words starting at address in $esp
x/2wd $esp Examine two 4-byte words starting at address in $esp
    in decimal
x/g $esp Examine 8-byte word starting at address in $esp.
x/gd $esp Examine 8-byte word starting at address in $esp
    in decimal
x/a $esp Examine address in $esp
    as offset from previous global symbol
x/s 0xbffff890 Examine string stored at 0xbffff890
x/20b sum Examine first 20 opcode bytes of function sum
x/10i sum Examine first 10 instructions of function sum
display /FMT EXPR Print expression EXPR using format FMT each time
    execution stops

Formats: x/[NUM][SIZE][FORMAT]
If not given, uses sensible default or last-used explicit format

NUM = number of objects to display
SIZE = size of each object
b = byte
h = half-word
w = word
q = qword (quad-word)

FORMAT = format for displaying each object
d = decimal
x = hex
o = octal