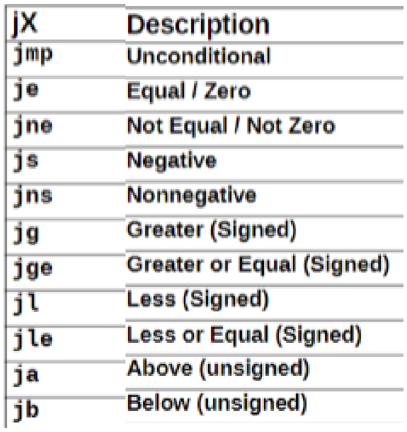
Assembly Guide

- **add** – add operands
- **sub** – subtract operands
- **and** – logical AND
- **or** – logical OR
- **xor** – logical XOR
- **cmp** – compare two operands
- **test** – compare two logical operands
- **Jumps**: usually follow a “cmp” or “test” command



jX	Description
jmp	Unconditional
je	Equal / Zero
jne	Not Equal / Not Zero
js	Negative
jns	Nonnegative
jg	Greater (Signed)
jge	Greater or Equal (Signed)
jl	Less (Signed)
jle	Less or Equal (Signed)
ja	Above (unsigned)
jb	Below (unsigned)

- **lea** – load effect address
  - Loads **address** of operand into second operand
- **mov**
  - loads **data** at address of operand into second operand
- **call** – call a procedure/function
- **ret** – return from procedure/function
- **push** – push operand onto stack
- **pop** – pop operand onto stack
- if you’re wondering what “l” and “q” mean:
  - **l** – long (word)– 32 bits
  - **q** – quad (quadword) – 64 bits

https://www.felixcloutier.com/x86/

GDB Guide

- **gdb** [executable name] – opens gdb on the given executable
  - gdb bomb
- **r** – runs the program
- **b [location]** – sets breakpoint – do this *before* you run
  - could be function name, address, line number
  - b phase_1
- **info r** – shows info and contents of the registers in use
- **x/(format) [address]** – accesses data from memory
  - x $eax
  - x 0xaaab580f
  - Format:
    - x/s $eax – prints string value
    - x/d $eax – prints int value
- **si** – steps into next instruction or function (if any)
- **ni** – next instruction
- **c** – continue running until end of program or next breakpoint
- **disas** – disassemble the code – shows the underlying assembly (this is your best friend)
- **d [breakpoint number]** – deletes breakpoint
- **q** – quits gdb

In terminal…

- **objdump -d [executable name]** – shows ALL assembly of the phases and supplementary functions that are used – could be useful but it is a lot of information at once

Other Notes

- if you accidentally reach the explode function but it doesn't actually print that you exploded, you are safe
  - you can quit gdb or rerun at this point
- if you finished many phases and you don't want to retype the answers, just store them in a solutions.txt file (write every phase's answer on a new line). Then in gdb, type "**r solutions.txt**" and you should be good. Only do this if you are comfortable with gdb and how to use it – you don't want to risk exploding.