GOTW1.1. GDB of the Week #1

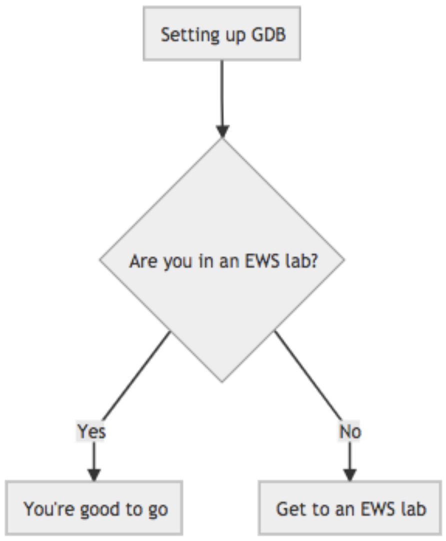
Week 1: Getting Started

1.1 What is GDB?

GDB (GNU Debugger) is a debugger. It let's you do more than print variables - you can call specific functions, change variables, stop the program, and lot's of other useful things you'll learn from this series. It works for languages like C and C++.

From now on, GDB is your best buddy (at least for this course). Because let's face it, using print statements as a debugger will only get you so far.

1.2 How Do I Setup GDB?



1.3 How Do I Use GDB?

An important note: From here on, I will assume that you are using a linux machine for simplicity.

First, you need to compile your programs with the `-g` flag. For example:

```
gdb@cs225> clang++ -stdlib=c++1y -stdlib=libc++ -g main.c -o main
```

Now to actually use the debugger. There are two ways to start up GDB:

1. `gdb` : Use gdb in interactive mode
2. `gdb [program name]` : Specify a program to load with gdb

Once you start GDB, your terminal will look like this:

```
(gdb) [insert commands here]
```

Example

```
gdb@cs225: gdb main
```

1.4 Useful Commands

GOTW 1

Total points:	0/1
Score:	0%

Question

Value:	1
History:	
Awarded points:	0/1

Report an error in this question

Previous question

Next question

Here are some useful commands to know about GDB. This will be at the top of every article, and we'll highlight the new commands. The highlighted commands will be the focus of the article.

On a side note, here is a more complete, condensed list of gdb commands.

Mini GDB Guide

help

```
(gdb) help
```

With `help`, you can see a list of topics that you can learn more about by doing.

You can learn more about a topic:

```
(gdb) help [topic name]
```

Or if you know the command:

```
(gdb) help [command]
```

file

Loads a program (by using the *program name*) to run gdb with.

```
(gdb) file [program name]
```

run

Starts the debugged program. Can also just use `r`.

```
(gdb) run
```

You can also run the program with arguments, as well as input/output redirection:

```
(gdb) run arg1 arg2 ...
```

- A neat trick:

Alternatively, you can also do:

```
gdb@cs225> gdb --args ./program_name arg1 arg2 ...
        (gdb) run
```

quit

Exit gdb.

```
(gdb) q
```

break (b)

```
(gdb) break [file]:[line number]
```

A breakpoint tells gdb to pause your program on `line number` in `file`.

When you reach a breakpoint, you can set another breakpoint in the same file by just specifying the `line number`.

```
(gdb) break [line number]
```

You can even set a breakpoint by function name. For example:

```
(gdb) b calcator.cpp:add
```

print

```
(gdb) print [variable | expression]
```

Print the value of the variable or expression. For example:

```
(gdb) print x
        $1 = 10
```

```
(gdb) print foo(5)
        $2 = 25
```

1.5 Task 1 - Hidden Traits

Now that we know some commands, let's try to get this debugger running.

Download gotw-01.sh

Using a terminal, extract the initial files by running the shell script you just downloaded (you will need to navigate to the directory where you saved the file):

```
sh gotw-01.sh
```

Your files for this problem will be in the `gotw-01` directory.

Here, we have provided you with a few files (`main.cpp`, `pokeman.cpp`, `pokeman.h`, and a `Makefile`).

Go ahead and run `make`, and try running the executable `main` with gdb.

You'll notice that there are some funky numbers on this mysterious Pokeman's stats. The question is: At what point of the code does the Pokeman's stats *actually* change?

Q1: On which code line number in main.cpp does the Pokeman mystery's stats change?

- ○ (a) 21
- ○ (b) 18
- ○ (c) 16
- ○ (d) 19

Q2: What is the value of the Pokeman squirrel's health at the end of main?

- ○ (a) 25
- ○ (b) 20
- ○ (c) 35
- ○ (d) 30

Q3: What is the value of the Pokeman charmanda's health when the program reaches line 15 in main.cpp?

- ◯ (a) 48
- ◯ (b) 50
- ◯ (c) 45
- ◯ (d) 40

1.6 Conclusion

In this lesson, we learned about:

- reasons to learn gdb
- how to find resources
- basic gdb commands
- running a program in debug mode with gdb

Save & Grade Save only