

I have been knowing about GDB for a while. However, I rarely tend to use it because it did not realize how useful and effective features the GDB has. I started to dive deep into the GDB world. GDB might look more difficult to use and understand than other IDE programs' debuggers but it is really accurate finding errors where happened. So far, I think learning GDB has been a huge weapon to my programming skill to go further.

When you tried to find errors from your program, it's much more intuitive being able to see your entire source code with highlighted problem areas. I realized that it could be preferred than other gdb programs.

What I found GDB is helpful from this program#1 is that it lets me know where I am going to. Once I was testing the adding and removing data with doubly linked list data structure for school information but it kept showing nothing stored in the nodes. With GDB, I tried to keep track of where the data were going and what functions were called. As I followed step by step, I could finally the reason. It was because I was calling the wrong function.

Another way I used GDB for was copy constructor for graph data structure. Certainly, I had made the mistake at a certain point in the program to think that a derived class would call its base class constructor

implicitly. Therefore, the program kept presenting an error which was segment fault. By GDB, I made a break point where the error might happen. Then, I found that the program was not going to the right line.

There was another thing GDB helped me was a derived class wrapper

function. I tried to call a method from the base class. However, the function I had intended to call was not called because the program was actually calling a completely different function. How I found what the problem was using GDB features, break point and print. (gdb) p or print "variable name" shows the value the variable currently has. Especially, when you want to verify the data you want to store, it's really helpful they actually come in or not. So far, GDB has really powerful debugger. I would probably use it for my entire programming life.