

Debugging Menggunakan GDB

- #apa itu gdb
- #gambaran dan panduan singkat
- #menggunakan perintah gdb untuk men-debug program c

1. Apa itu GDB

GDB adalah singkatan dari "GNU Debugger." Ini adalah perangkat lunak debugger yang digunakan untuk menganalisis program baik/jahat dan memperbaiki bug dalam program-program yang ditulis dalam bahasa pemrograman seperti C/C++, Assembly dan Fortran.

2. Gambaran dan Panduan singkat

- #install tools gdb, kalo di windows menggunakan MinGW dan kalo di debian/ubuntu dan turunannya { apt install gdb -y }
- #buat program c yang minimalis menggunakan sublime text
- #compile menggunakan perintah ini, { gcc -g program.c -o program }, perintah tersebut menginstruksikan compiler GCC untuk mengompilasi kode sumber dari file "program.c", menyertakan informasi debugging, dan menghasilkan file eksekusi dengan nama "program".
- #cek sudahkah berhasil ter compile atau menjadi file executeble dengan perintah, { ls -l program }
- #buka di sublime text hasil file yang telah dikompilasi menjadi kode mesin/biner/binary
- #cek dengan perintah { file program } untuk mengetahui informasi dari suatu file

3

- #lanjut gunakan perintah ini untuk mendebug { gdb ./program }
- #ketik { run } untuk menjalankan programnya
- #ketik { list } untuk melihat program c, meskipun sudah menjadi kode mesin/binary gdb memiliki kemampuan untuk me-reverse kode mesin kembali menjadi kode aslinya, dengan catatan semakin kompleks programnya, akan semakin jauh mendekati penulisan program aslinya, sedangkan dari segi fungsionalitas tetap sama, sesuai program yang dibuat kaya gimana.
- #ketik { break main }, perintah tersebut berarti menghentikan eksekusi program saat mencapai fungsi `main` untuk menganalisis dan debug.
- #kemudian ketik { run }, akan di tampilkan spesifik lokasi pada fungsi main
- #ketik { next } cek baris per baris kode
- #ketik { print variable-nya }, perintah ini digunakan untuk mencetak nilai variabel atau ekspresi selama debugging.
- #ketik { display nilai-nya }, perintah ini digunakan untuk secara otomatis mencetak nilai ekspresi setiap kali program berhenti dalam debugging.
- { help display, info display, undisplay nilainya }
- #ketik { quit }, untuk mengahiri proses debug

ada banyak yang saya tidak share, karena memang masih pemula dan pengalaman belum banyak, contoh beberapa ==> layout split | gcc -m32 -o output_program input_program.c | dan lain-lain <=< bila ada yang kurang/tidak mengerti bisa di tanyakan di grup reversingid