



Laboratorio di sistemi operativi – T4

Debugging di un programma con `gdb`

GDB: the GNU Project debugger

- Software che permette di effettuare un debug di un programma
 - Definizione di breakpoint
 - Esecuzione step-by-step
 - Controllo di un valore di una variabile/espressione
 - Gestione dello stack
 - Gestione dei threads
 - ...
- <https://www.gnu.org/software/gdb/gdb.html>

Eseguire gdb

- Prerequisiti: compilare il programma con i flags
 - `-g` , per aggiungere informazioni aggiuntive utili al debug
 - `-O0` per disabilitare tutte le ottimizzazioni del codice
- Per eseguire un programma con gdb

```
gcc -g -O0 programma.c
```

```
gdb programma
```

- Non inserire eventuali parametri del programma, verranno inseriti successivamente

I comandi gdb

- Dopo la sua esecuzione, appare il prompt
 - (gdb)
- A questo punto, I comandi di gdb possono essere inseriti
 - `help` elenca I comandi a disposizione
 - `list` mostra una porzione del codice sorgente del programma
 - `list <numero-linea>` mostra il codice sorgente intorno alla linea indicata in `<numero-linea>`
 - `break <numero-linea>` inserisce un breakpoint alla linea `<numero-linea>`
 - `break <nomefile>:<numero-linea>`
 - `info b` mostra i breakpoint definiti (ogni breakpoint è definito da un `<id>`)
 - `del <id>` elimina un breakpoint

I comandi gdb

- `run` , esegue il programma
 - `run <parametri>` esegue il programma con i parametri specificati
- `next` esegue una linea di codice. Se si tratta di una funzione, la esegue
- `step` esegue una linea di codice. Se si tratta di una funzione, entra nel codice della funzione
- `cont` prosegue l'esecuzione del programma fino al prossimo breakpoint
- `print <espressione>` valuta l'espressione e ne stampa il valore
- `display <espressione>` valuta l'espressione e ne stampa il valore ad ogni step
- `bt` mostra lo stack delle invocazioni di funzione (backtrace)
- `quit` termina l'esecuzione di gdb

Fork e gdb

- E' possibile connettere gdb ad un processo in esecuzione
 - `gdb -p <PID>`
 - Se il processo sta eseguendo una system call, potrebbe essere necessario eseguire come superuser con `sudo`
 - `sudo gdb -p <PID>`
- Domanda: se sto facendo debug di un processo che successivamente esegue una `fork`, quale dei due processi sarà oggetto di debug?
 - In gdb utilizzare uno dei due comandi
 - `set follow-fork-mode parent`
 - `set follow-fork-mode child`