

Progetto e realizzazione di una estensione VSCode per il debugging di un nucleo multiprogrammato

Tesi di Laurea in Ingegneria Informatica

Candidato

Francesco Mignone

Relatori

Prof. Giuseppe Lettieri

Prof. Luigi Leonardi



UNIVERSITÀ DI PISA

Introduzione e Problema

- Il nucleo multiprogrammato di un sistema operativo è il componente fondamentale che gestisce le risorse hardware e software. Gli studenti che si avvicinano alla programmazione del nucleo necessitano di strumenti per il debug.
- Ma gli strumenti per il debug richiedono una curva di apprendimento molto elevata e talvolta l'interfaccia proposta è molto complicata:
 - Come possiamo rendere più facile l'approccio al debug?
 - È possibile rendere unica e semplice l'interfaccia di debug?

- Integrare l'ambiente di debug all'interno di un interfaccia e applicazione già conosciuta dall'utente, VSCode:
 - Implementa un interfaccia di debug generica
 - Permette lo sviluppo di estensioni personalizzate

- Estendere le funzionalità di base del debugger per fornire funzionalità e informazioni aggiuntive sullo stato del nucleo
 - Processi in esecuzione
 - Informazioni sulla Memoria Virtuale
 - Etc...

Risultato

The screenshot shows the GDB debugger interface with the following components:

- Top Panel:** Menu bar (File, Edit, Selection, View, Go, Run, Terminal, Help) and toolbar.
- Left Sidebar:**
 - VARIABLES:** Shows 'p' at address 0x12c0.
 - WATCH:** Shows 'p->barrier_id = 2'.
 - CALL STACK:** Shows the current call stack with 'check_barrier(des_proc * p)' at the top.
 - BREAKPOINTS:** Shows a breakpoint at line 2152 of 'sistema.cpp'.
- Main Window:** Displays the source code of 'sistema.cpp'. The current line is 2152: `barrier_t *b = &barriers[p->barrier_id];`. The code includes a `c_barrier_create` function and a `check_barrier` function.
- Right Sidebar:**
 - PROCESSI IN ESECUZIONE:** Shows details for process 'sistema', including its PID (3), level, and registers.
- Bottom Panel:** Shows the GDB console output, including a warning about the target architecture and breakpoint information.