ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

CS255 - Programming Lab

Ενότητα: Tutorials

Άγγελος Μπίλας

Τμήμα Επιστήμης Υπολογιστών

Tutorial 4 - GDB more advanced

Breakpoints / Watchpoints / Catchpoints

Breakpoints stop the program execution when a specific code segment is reached.

Catchpoints stop the program execution at specific events (e.g., syscalls, fork etc.).

Watchpoints stop the program execution when a specific memory segment is accessed. One can enable a watchpoint to stop the program execution only at writes watch, reads rwatch or both accesses awatch.

Taking action when the program stops

If you want gdb to do something at breaks you can use command, i.e.:

(gdb) break main.c:43

Breakpoint 1 at 0x7fffe234: file main.c, line 43.

(gdb) command

Type commands for breakpoint(s) 1, one per line.

End with a line saying just "end".

>silent

>print x

>end

(gdb)

Examine Memory

To examine a memory segment you can use x[/FMT] ADDRESS.

ADDRESS is an expression for the memory address to examine.
FMT is a repeat count followed by a format letter and a size letter.
Format letters are o(octal), x(hex), d(decimal), u(unsigned decimal), t(binary), f(float), a(address), i(instruction), c(char), s(string) and z(hex, zero padded on the left).

Example 1

(gdb) x &i

0x7fffffffe578: 0x0000000f

This example prints the contents of the memory at address &i (essentially i contents).

Example 2

(gdb) x/d &i

0x7fffffffe578: 15

This example prints the contents of the memory at address &i (essentially i contents) in decimal format.

Example 3

(gdb) x/zb &i

0x7fffffffe578: 0x0f

This example prints the contents of a single byte at address &i in hexadecimal format padded with zeroes on the left.

Example 4

(gdb) x/4b &i

0x7fffffffe578: 0x0f 0x00 0x00 0x00

This example prints the contents of the four bytes at address &i in hexadecimal format padded with zeroes on the left.

Example 5

(gdb) x/4w &i

0x7fffffffe578: 0x0000000f 0x00000000 0x00400570 0x00000000

This example prints the contents of the four words at address &i in hexadecimal format padded with zeroes on the left.

Example 6

```
(gdb) x/10i $pc
=> 0x40052b <main+37>:  mov    $0x4005f4,%esi
   0x400530 <main+42>:  mov    $0x4005f9,%edi
   0x400535 <main+47>:  mov    $0x0,%eax
   0x40053a <main+52>:  callq  0x4003e0 <printf@plt>
   0x40053f <main+57>:  mov    $0x4005f4,%esi
   0x400544 <main+62>:  mov    $0x400600,%edi
   0x400549 <main+67>:  mov    $0x0,%eax
   0x40054e <main+72>:  callq  0x4003e0 <printf@plt>
   0x400553 <main+77>:  mov    $0x80,%esi
   0x400558 <main+82>:  mov    $0x400608,%edi
```

This example prints 10 instructions starting from the current address of the program counter ($pc).

Modify memory contents

One can change the memory contents using set.

Example

```
(gdb) info locals
i = 15
j = 5
(gdb) set j = 0
(gdb) info locals
i = 15
j = 0
```

Moving in the stack

One can move up or down the stack using the up and down commands respectively. To directly jump at a specific stack frame you can use select-frame.

References:

- http://www.gnu.org/software/gdb/

Authored by: Foivos S. Zakkak

Άδειες Χρήσης

•Το παρόν εκπαιδευτικό υλικό υπόκειται στην άδεια χρήσης Creative Commons και ειδικότερα

Αναφορά – Μη εμπορική Χρήση – Όχι Παράγωγο Έργο 3.0 Ελλάδα

(Attribution – Non Commercial – Non-derivatives 3.0 Greece)

•Εξαιρείται από την ως άνω άδεια υλικό που περιλαμβάνεται στις διαφάνειες του μαθήματος, και υπόκειται σε άλλου τύπου άδεια χρήσης. Η άδεια χρήσης στην οποία υπόκειται το υλικό αυτό αναφέρεται ρητώς.

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.