megapixel@megapixel-VirtualBox:~/Рабочий стол$ gcc -std=c99 -pedantic -Wall -ggdb prog.c

megapixel@megapixel-VirtualBox:~/Рабочий стол$ gdb ./a.out

GNU gdb (Ubuntu 9.2-0ubuntu1~20.04) 9.2

Copyright (C) 2020 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

Type "show copying" and "show warranty" for details.

This GDB was configured as "x86\_64-linux-gnu".

Type "show configuration" for configuration details.

For bug reporting instructions, please see:

<http://www.gnu.org/software/gdb/bugs/>.

Find the GDB manual and other documentation resources online at:

<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".

Type "apropos word" to search for commands related to "word"...

Reading symbols from ./a.out...

(gdb) help

List of classes of commands:

aliases -- Aliases of other commands.

breakpoints -- Making program stop at certain points.

data -- Examining data.

files -- Specifying and examining files.

internals -- Maintenance commands.

obscure -- Obscure features.

running -- Running the program.

stack -- Examining the stack.

status -- Status inquiries.

support -- Support facilities.

tracepoints -- Tracing of program execution without stopping the program.

user-defined -- User-defined commands.

Type "help" followed by a class name for a list of commands in that class.

Type "help all" for the list of all commands.

Type "help" followed by command name for full documentation.

Type "apropos word" to search for commands related to "word".

Type "apropos -v word" for full documentation of commands related to "word".

Command name abbreviations are allowed if unambiguous.

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

38

[Inferior 1 (process 3226) exited normally]

(gdb) list

1 #include <stdio.h>

2

3 int rec(int a){

4 if (a <= 0) {

5 return 0;

6 }

7 return a + rec(a-1) \* (rec(a-3) + 5);

8 }

9

10 int proc(int a){

(gdb) list

11 return a + 2;

12 }

13

14 int main() {

15 int a = 3;

16

17 a = proc(a);

18

19 printf("%d\n", rec(a - 2));

20

(gdb) list rec

1 #include <stdio.h>

2

3 int rec(int a){

4 if (a <= 0) {

5 return 0;

6 }

7 return a + rec(a-1) \* (rec(a-3) + 5);

8 }

9

10 int proc(int a){

(gdb) break 15

Breakpoint 1 at 0x11b5: file prog.c, line 15.

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, main () at prog.c:15

15 int a = 3;

(gdb) print a

$1 = 0

(gdb) break 16

Breakpoint 2 at 0x5555555551bc: file prog.c, line 17.

(gdb) run

The program being debugged has been started already.

Start it from the beginning? (y or n) y

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, main () at prog.c:15

15 int a = 3;

(gdb) next

Breakpoint 2, main () at prog.c:17

17 a = proc(a);

(gdb) print a

$2 = 3

(gdb) next

19 printf("%d\n", rec(a - 2));

(gdb) print a

$3 = 5

(gdb) ptype a

type = int

(gdb) set args "Hello, World"

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out "Hello, World"

38

[Inferior 1 (process 3312) exited normally]

(gdb) show args

Argument list to give program being debugged when it is started is ""Hello, World"".

(gdb)

(gdb) break 17

Breakpoint 1 at 0x11bc: file prog.c, line 17.

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, main () at prog.c:17

17 a = proc(a);

(gdb) continue

Continuing.

38

[Inferior 1 (process 3347) exited normally]

(gdb)

(gdb) b 16

Breakpoint 1 at 0x11bc: file prog.c, line 17.

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, main () at prog.c:17

17 a = proc(a);

(gdb) print a

$1 = 3

(gdb) set var a =5

(gdb) print a

$2 = 5

(gdb)

**Разница между next и step**

(gdb) b 16

Breakpoint 1 at 0x11bc: file prog.c, line 17.

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, main () at prog.c:17

17 a = proc(a);

(gdb) next

19 printf("%d\n", rec(a - 2));

(gdb) next

38

21 return 0;

(gdb) next

22 }

(gdb) run

The program being debugged has been started already.

Start it from the beginning? (y or n) y

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, main () at prog.c:17

17 a = proc(a);

(gdb) step

proc (a=21845) at prog.c:10

10 int proc(int a){

(gdb) step

11 return a + 2;

(gdb) step

12 }

(gdb) step

main () at prog.c:19

19 printf("%d\n", rec(a - 2));

(gdb) step

rec (a=21845) at prog.c:3

3 int rec(int a){

(gdb) step

4 if (a <= 0) {

(gdb) step

7 return a + rec(a-1) \* (rec(a-3) + 5);

(gdb) step

rec (a=0) at prog.c:3

3 int rec(int a){

(gdb) step

4 if (a <= 0) {

(gdb) step

7 return a + rec(a-1) \* (rec(a-3) + 5);

(gdb) step

rec (a=32767) at prog.c:3

3 int rec(int a){

(gdb) step

4 if (a <= 0) {

(gdb) step

7 return a + rec(a-1) \* (rec(a-3) + 5);

(gdb) step

rec (a=0) at prog.c:3

3 int rec(int a){

(gdb) step

4 if (a <= 0) {

(gdb) step

5 return 0;

(gdb) step

8 }

(gdb) step

rec (a=0) at prog.c:3

3 int rec(int a){

(gdb) step

4 if (a <= 0) {

(gdb) step

5 return 0;

(gdb) step

8 }

(gdb) step

rec (a=1) at prog.c:7

7 return a + rec(a-1) \* (rec(a-3) + 5);

(gdb) step

8 }

(gdb) quit

megapixel@megapixel-VirtualBox:~/Рабочий стол$

**Бесконечная рекурсия**

(gdb) list

1 #include <stdio.h>

2

3 int rec(int a){

4 if (a == 0) {

5 return 0;

6 }

7 return a + rec(a-1) \* (rec(a-3) + 5);

8 }

9

10 int proc(int a){

(gdb) list

11 return a + 2;

12 }

13

14 int main() {

15 int a = 3;

16

17 a = proc(a);

18

19 printf("%d\n", rec(-a - 2));

20

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

Program received signal SIGSEGV, Segmentation fault.

0x000055555555516e in rec (a=-174681) at prog.c:7

7 return a + rec(a-1) \* (rec(a-3) + 5);

(gdb) bt

#0 0x000055555555516e in rec (a=-174681) at prog.c:7

#1 0x0000555555555173 in rec (a=-174680) at prog.c:7

#2 0x0000555555555173 in rec (a=-174679) at prog.c:7

#3 0x0000555555555173 in rec (a=-174678) at prog.c:7

#4 0x0000555555555173 in rec (a=-174677) at prog.c:7

#5 0x0000555555555173 in rec (a=-174676) at prog.c:7

#6 0x0000555555555173 in rec (a=-174675) at prog.c:7

#7 0x0000555555555173 in rec (a=-174674) at prog.c:7

#8 0x0000555555555173 in rec (a=-174673) at prog.c:7

#9 0x0000555555555173 in rec (a=-174672) at prog.c:7

#10 0x0000555555555173 in rec (a=-174671) at prog.c:7

#11 0x0000555555555173 in rec (a=-174670) at prog.c:7

#12 0x0000555555555173 in rec (a=-174669) at prog.c:7

#13 0x0000555555555173 in rec (a=-174668) at prog.c:7

#14 0x0000555555555173 in rec (a=-174667) at prog.c:7

#15 0x0000555555555173 in rec (a=-174666) at prog.c:7

#16 0x0000555555555173 in rec (a=-174665) at prog.c:7

#17 0x0000555555555173 in rec (a=-174664) at prog.c:7

#18 0x0000555555555173 in rec (a=-174663) at prog.c:7

#19 0x0000555555555173 in rec (a=-174662) at prog.c:7

#20 0x0000555555555173 in rec (a=-174661) at prog.c:7

#21 0x0000555555555173 in rec (a=-174660) at prog.c:7

#22 0x0000555555555173 in rec (a=-174659) at prog.c:7

#23 0x0000555555555173 in rec (a=-174658) at prog.c:7

#24 0x0000555555555173 in rec (a=-174657) at prog.c:7

#25 0x0000555555555173 in rec (a=-174656) at prog.c:7

--Type <RET> for more, q to quit, c to continue without paging--q

Quit

**Обычная рекурсия**

(gdb) list

1 #include <stdio.h>

2

3 int rec(int a){

4 if (a <= 0) {

5 return 0;

6 }

7 return a + rec(a-1) \* (rec(a-3) + 5);

8 }

9

10 int proc(int a){

(gdb) b 5

Breakpoint 1 at 0x115f: file prog.c, line 5.

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, rec (a=0) at prog.c:5

5 return 0;

(gdb) bt

#0 rec (a=0) at prog.c:5

#1 0x0000555555555173 in rec (a=1) at prog.c:7

#2 0x0000555555555173 in rec (a=2) at prog.c:7

#3 0x0000555555555173 in rec (a=3) at prog.c:7

#4 0x00005555555551d6 in main () at prog.c:19

(gdb)

**Косвенная рекурсия**

(gdb) list f2

1 #include <stdio.h>

2

3 int f1(int a);

4

5 int f2(int a) {

6 if (a <= 0) {

7 return 1;

8 }

9

10 return a+f1(a-1);

(gdb) list f1

8 }

9

10 return a+f1(a-1);

11 }

12

13 int f1(int a) {

14 if (a <= 0) {

15 return 1;

16 }

17

(gdb) b 15

Breakpoint 1 at 0x118e: file prog.c, line 15.

(gdb) run

Starting program: /home/megapixel/Рабочий стол/a.out

Breakpoint 1, f1 (a=0) at prog.c:15

15 return 1;

(gdb) bt

#0 f1 (a=0) at prog.c:15

#1 0x0000555555555172 in f2 (a=1) at prog.c:10

#2 0x00005555555551a2 in f1 (a=2) at prog.c:18

#3 0x0000555555555172 in f2 (a=3) at prog.c:10

#4 0x00005555555551a2 in f1 (a=4) at prog.c:18

#5 0x0000555555555172 in f2 (a=5) at prog.c:10

#6 0x00005555555551a2 in f1 (a=6) at prog.c:18

#7 0x0000555555555172 in f2 (a=7) at prog.c:10

#8 0x00005555555551a2 in f1 (a=8) at prog.c:18

#9 0x0000555555555172 in f2 (a=9) at prog.c:10

#10 0x00005555555551a2 in f1 (a=10) at prog.c:18

#11 0x00005555555551c6 in main () at prog.c:25

(gdb)