

שאלה 1

הרצה של הפונקציה הרקורסיבית recursive_fun(100000) על מנת ליצור גלישה מהמחשנית. ע"י ריצה ללא דגל -ggdb-

```
ariela@ariela-VirtualBox:~/Desktop/opSys_1$ make ex_1_no_debug
gcc -o ex_1 ex1.c
ariela@ariela-VirtualBox:~/Desktop/opSys_1$ make core
sudo sysctl -w kernel.core_pattern=core && \
sudo sysctl -w fs.suid_dumpable=1 && \
bash -c 'ulimit -c unlimited && ./ex_1'
[sudo] password for ariela:
kernel.core_pattern = core
fs.suid_dumpable = 1
Segmentation fault (core dumped)
make: *** [makefile:17: core] Error 139
```

יצר קובץ core, מראה שיש איזשהי שגיאה של זיכרון.

```
For help, type "help".
Type "apropos word" to search for commands related to "word"...
--Type <RET> for more, q to quit, c to continue without paging--
Reading symbols from ex_1...

This GDB supports auto-downloading debuginfo from the following URLs:
  <https://debuginfod.ubuntu.com>
Enable debuginfod for this session? (y or [n])
Debuginfod has been disabled.
To make this setting permanent, add 'set debuginfod enabled off' to .gdbinit.
(No debugging symbols found in ex_1)
[New LWP 3661]
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Core was generated by './ex_1'.
Program terminated with signal SIGSEGV, Segmentation fault.
#0  0x0000640dbb183135 in recursive_fun ()
(gdb)
```

ניתן לראות רק באיזה פונקציה נוצרה השגיאה.

```
ariela@ariela-VirtualBox:~/Desktop/opSys_1$ make ex_1
gcc -o ex_1 ex1.c -ggdb
ariela@ariela-VirtualBox:~/Desktop/opSys_1$ make core
sudo sysctl -w kernel.core_pattern=core && \
sudo sysctl -w fs.suid_dumpable=1 && \
bash -c 'ulimit -c unlimited && ./ex_1'
kernel.core_pattern = core
fs.suid_dumpable = 1
Segmentation fault (core dumped)
make: *** [makefile:17: core] Error 139
ariela@ariela-VirtualBox:~/Desktop/opSys_1$
```

הרצאה זהה אבל עם הדגל.

```
Reading symbols from ex_1...
[New LWP 3980]

This GDB supports auto-downloading debuginfo from the following URLs:
  <https://debuginfod.ubuntu.com>
Enable debuginfod for this session? (y or [n])
Debuginfod has been disabled.
To make this setting permanent, add 'set debuginfod enabled off' to .gdbinit.
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Core was generated by `./ex_1'.
Program terminated with signal SIGSEGV, Segmentation fault.
#0  0x00005a76de8e814d in recursive_fun (n=738181) at ex1.c:9
9      return n * recursive_fun(n - 1);
(gdb)
```

מראה לא רק את הפונקציה בה נוצרת השגיאה, אלא גם באיזה שורה זה נפל ולמה.

```
ex1.c
2
3 // 1.1
4 //recursive function that exceeds from stack memory
5 int recursive_fun(int n) {
6     if (n == 0) {
7         return 1;
8     }
9     return n * recursive_fun(n - 1);
10 }
11
12 // 1.2
13 //deviation by 0 [!=0]
14 int division(int a, int b) {
15     return a / b;
16 }
17
18 // 1.3

multi-thre No process (src) In: L?? PC: ??
Breakpoint 1, main () at ex1.c:32
(gdb) next

Program received signal SIGSEGV, Segmentation fault.
0x0000555555555135 in recursive_fun (n=<error reading variable: Cannot access memory at address 0x7ffff7feffc>) at ex1.c:5
(gdb) next

Program terminated with signal SIGSEGV, Segmentation fault.
The program no longer exists.
(gdb)
```

השתמשנו ב -tui בתור דבאגר גרפי.