

# Lista de Exercícios - Estrutura de Dados

## 1 Código 1 - Fibonacci

**Enunciado:** a) Depure o código para descobrir quantas vezes o fibonacci(5) é calculado (requisitado).

**Solução:** 89 vezes. Seguindo os passos a seguir:

---

```
(gdb) break 7 if n == 5
Breakpoint 1 at 0x1144: file fibonacci.c, line 7.
(gdb) ignore 1 10000
Will ignore next 10000 crossings of breakpoint 1.
(gdb) continue
The program is not being run.
(gdb) run
[Inferior 1 (process 15758) exited normally]
(gdb) info breakpoints
Num      Type             Disp Enb Address              What
1        breakpoint     keep y   0x0000555555555144  in fibonacci at
        fibonacci.c:7
        stop only if n == 5
        breakpoint already hit 89 times
        ignore next 9911 hits
(gdb)
```

---

**Enunciado:** b) Descubra através da depuração, qual a maior quantidade de níveis (frames) esse código alcança. **Solução:** 8 frames.

---

```
(gdb) backtrace
#0 fibonacci (n=2) at fibonacci.c:7
#1 0x0000555555555151 in fibonacci (n=3)
    at fibonacci.c:7
#2 0x0000555555555160 in fibonacci (n=5)
    at fibonacci.c:7
#3 0x0000555555555160 in fibonacci (n=7)
    at fibonacci.c:7
#4 0x0000555555555160 in fibonacci (n=9)
    at fibonacci.c:7
#5 0x0000555555555160 in fibonacci (n=11)
    at fibonacci.c:7
#6 0x0000555555555160 in fibonacci (n=13)
    at fibonacci.c:7
#7 0x0000555555555160 in fibonacci (n=15)
```

```
at fibonacci.c:7
#8 0x00005555555517e in main () at fibonacci.c:13
```

---

**Enunciado:** b) Descubra através da depuração, qual a maior quantidade de níveis (frames) esse código alcança. **Solução:** 8 frames.

---

```
(gdb) break 7 if n == 14
Breakpoint 1 at 0x1144: file fibonacci.c, line 7.
(gdb) run
Starting program: /home/adrian/UFERSA/III/Algoritmos e Estrutura
de Dados II - PEX1246/aedii/lab/lista1/fibonacci
Breakpoint 1, fibonacci (n=14) at fibonacci.c:7
7      return fibonacci(n - 1) + fibonacci(n - 2);
(gdb) step
fibonacci (n=13) at fibonacci.c:3
3      if (n <= 1) {
(gdb) c
Continuing.
[Inferior 1 (process 16188) exited normally]
(gdb) run
Breakpoint 1, fibonacci (n=14) at fibonacci.c:7
7      return fibonacci(n - 1) + fibonacci(n - 2);
(gdb) step
fibonacci (n=13) at fibonacci.c:3
3      if (n <= 1) {
(gdb)
```

---

## 2 Código 2 - Operação Matemática

**Enunciado:** **Solução:** Escreva a solução do exercício aqui.

**Solução:**

---

```
(gdb) break opmat
Function "opmat" not defined.
Make breakpoint pending on future shared library load? (y or [n]) n
(gdb) break operacaoMatematica
Breakpoint 1 at 0x1197: file opmat.c, line 5.
(gdb) run
Digite o primeiro numero: 10
Digite o segundo numero: 20

Breakpoint 1, operacaoMatematica (a=10, b=20)
```

```

    at opmat.c:5
5      while (b != 0) {
(gdb)

[Thread debugging using libthread_db enabled]
Using host libthread_db library
    "/lib/x86_64-linux-gnu/libthread_db.so.1".
Digite o primeiro numero: 10
Digite o segundo numero: 20

Breakpoint 1, operacaoMatematica (a=10, b=20)
    at opmat.c:5
5      while (b != 0) {

```

---

**Enunciado: Solução:** b) Descubra através da depuração, qual a maior profundidade de (frames) esse código alcança.

---

```

(gdb) backtrace
#0 operacaoMatematica (a=10, b=20) at opmat.c:5
#1 0x0000555555555242 in main () at opmat.c:21
(gdb)

```

---

**Enunciado: Solução:** c) Descubra o que ocorre quando um parâmetro é 0.

---

```

(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/adrian/UFERSA/III/Algoritmos e Estrutura
    de Dados II - PEX1246/aedii/lab/lista1/opmat
[Thread debugging using libthread_db enabled]
Using host libthread_db library
    "/lib/x86_64-linux-gnu/libthread_db.so.1".
Digite o primeiro numero: 10
Digite o segundo numero: 0

Breakpoint 1, operacaoMatematica (a=10, b=0)
    at opmat.c:5
5      while (b != 0) {
(gdb) c
Continuing.
0 resultado da operacao entre 10 e 0 : 10
[Inferior 1 (process 16551) exited normally]

```

---

**Enunciado: Solução:** d) Descubra o que ocorre quando os dois parâmetros apresentados são iguais e maiores que 1.

---

```
library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Digite o primeiro numero: 10
Digite o segundo numero: 10

Breakpoint 1, operacaoMatematica (a=10, b=10)
  at opmat.c:5
5      while (b != 0) {
(gdb) next
6          r = a % b;
(gdb) next
7          a = b;
(gdb) next
8          b = r;
(gdb) next
5      while (b != 0) {
(gdb) next
10     return a;
(gdb) next
11 }
(gdb) next
main () at opmat.c:23
23     printf("0 resultado da operacao entre %d e %d : %d\n", num1,
    num2, res);
(gdb) next
0 resultado da operacao entre 10 e 10 : 10
25     return 0;
```

---

### 3 Código 3 - Numeros Primos

**Enunciado:** Descreva o enunciado do exercício aqui.

**Solução:** Escreva a solução do exercício aqui.