

Aluno: Wanderson A. de Oliveira

LINK PARA REPOSITÓRIO GIT

<https://github.com/wanderson130/LAB1>

RECOMENDAÇÕES PARA COMPILAR O PROGRAMA

- 1 – Use o comando make para gerar os executáveis
- 2 – acesse a pasta bin através da linha de comando
- 3 – utilize o comando ./geometrica caso queira executar o programa da questao 1
- 4 – utilize o comando ./anterior <numero inteiro> para executar o programa da questao 2

COMENTÁRIOS SOBRE AS QUESTÕES 2 E 3 DO LAB1 - LP1

2) Como podemos ver a partir da utilização do gprof, a função Primo é chamada 8 vezes, sendo assim a de maior tempo de execução.

```
wanderson.alves@A305-03L:~/Documentos/q2/bin$ gprof --brief -p anterior
Flat profile:

Each sample counts as 0.01 seconds.
no time accumulated

%   cumulative   self           calls   self   total    name
time   seconds    seconds                Ts/call Ts/call  name
-----
0.00      0.00      0.00             8      0.00   0.00   Primo(int, int)
0.00      0.00      0.00             1      0.00   0.00   _GLOBAL__sub_I_Z5Primoii
0.00      0.00      0.00             1      0.00   0.00   _GLOBAL__sub_I_Z8fatorial
d
0.00      0.00      0.00             1      0.00   0.00   _GLOBAL__sub_I_main
0.00      0.00      0.00             1      0.00   0.00   maiorPrimo(int)
0.00      0.00      0.00             1      0.00   0.00   __static_initialization_an
d_destruction_0(int, int)
0.00      0.00      0.00             1      0.00   0.00   __static_initialization_an
d_destruction_0(int, int)
0.00      0.00      0.00             1      0.00   0.00   __static_initialization_an
d_destruction_0(int, int)
0.00      0.00      0.00             1      0.00   0.00   fatorial(double)
```

3) Utilizando o GDB conseguimos notar que nenhum dos valores de fato é alterado, pois eles são apenas utilizados internamente nas funções, mas como não fazemos as variáveis receberem um valor, nem utilizamos seu endereço para modificá-las, assim que saímos da função os valores continuam sendo os mesmos.

```
Breakpoint 1, main (argc=1, argv=0x7fffffffdd18) at q3.cpp:26
26      funcx (arg1, arg2);
(gdb) p arg1
$1 = 11
(gdb) p arg2
$2 = 23
(gdb) n

Breakpoint 2, main (argc=1, argv=0x7fffffffdd18) at q3.cpp:27
27      funcy (&arg1, arg2);
(gdb) p arg1
$3 = 11
(gdb) p arg2
$4 = 23
(gdb) n
28      int resultado = 0;
(gdb) p arg1
$5 = 11
(gdb) p arg2
$6 = 23
(gdb) n

Breakpoint 3, main (argc=1, argv=0x7fffffffdd18) at q3.cpp:29
29      funcz (arg1, arg2, &resultado);
(gdb) p arg1
$7 = 11
(gdb) p arg2
$8 = 23
(gdb) n

Breakpoint 4, main (argc=1, argv=0x7fffffffdd18) at q3.cpp:30
30      return 0;
(gdb) p arg1
$9 = 11
(gdb) p arg2
$10 = 23
(gdb) █
```