

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Sistemas Operacionais

Alunos: Anderson Alves, Leonardo Marques, Luanna Rabelo, Plínio Marini

Atividade 3 - Relatório

Professor: Marcelo Zanchetta do Nascimento

1. Exercício 6

a. Descrição

O programa foi resolvido utilizando três funções, das quais uma calcula o valor médio da função, uma que encontra o maior valor entre os números digitados e outra função que calcula o menor valor entre os números digitados. Além disso, foi utilizada uma função “obtemNumeros” que pega os valores passados por linha de comando em valores inteiros, colocando-os em um vetor, criando assim a lista de valores que serão utilizados pelas funções.

b. Diagrama de tempo de execução

```
[root@localhost ~]# ls
bench.py hello.c primos primos.c stats.c
[root@localhost ~]# gcc -o stats stats.c -pthread
[root@localhost ~]# strace -c ./stats 10 12 13 14 15
0 valor mínimo é 10
0 valor máximo é 15
0 valor médio é 12
% time      seconds  usecs/call   calls    errors syscall
-----
50.71      0.003600      720         5         0  openat
11.27      0.000800      400         2         0  munmap
 7.04      0.000500      100         5         0  fstat
 7.03      0.000499      33          15         0  mmap
 4.24      0.000301      100         3         0  read
 4.23      0.000300      100         3         0  clone
 4.23      0.000300      37          8         0  mprotect
 4.21      0.000299      299         1         1  faccessat
 2.82      0.000200      28          7         0  rt_sigprocmask
 1.41      0.000100      20          5         0  close
 1.41      0.000100      100         1         0  futex
 1.41      0.000100      100         1         0  set_robust_list
 0.00      0.000000      0           1         0  set_tid_address
 0.00      0.000000      0           2         0  rt_sigaction
 0.00      0.000000      0           1         0  uname
 0.00      0.000000      0           4         0  brk
 0.00      0.000000      0           1         0  execve
 0.00      0.000000      0           1         0  prlimit64
-----
100.00      0.007099              66         1 total
[root@localhost ~]#
```

c. Saídas gerada pelo programa

```
luanna@DESKTOP-H7LHBMV: ~/Trabalho3
luanna@DESKTOP-H7LHBMV:~/Trabalho3$ gcc stats.c -o stats -lpthread
luanna@DESKTOP-H7LHBMV:~/Trabalho3$ ./stats 22 31 45 65
0 valor médio é 40
0 valor mínimo é 22
0 valor máximo é 65
```

2. Exercício 7

a. Descrição

O programa foi feito por meio de uma função que verifica se o número digitado na linha de comando é um número primo ou não. Além dessa existe outra função chamada “produzirPrimos”, que é a função responsável pela produção dos números primos menores ou iguais ao número digitado inicialmente. Os números primos são identificados pela função “verificaPrimo” que identifica quais são os números que devem ser separados.

b. Diagrama de tempo de execução

```

bench.py hello.c primos primos.c
[root@localhost ~]# strace -c ./primos
Insira o numero: 37
37 31 29 23 19 17 13 11 7 5 3 2
% time      seconds  usecs/call   calls   errors syscall
-----
60.00      0.001200      240          5         0  openat
15.00      0.000300      150          2         0  munmap
10.00      0.000200       15         13         0  mmap
 5.00      0.000100      100          1         1  lseek
 5.00      0.000100       25          4         0  read
 5.00      0.000100       14          7         0  fstat
 0.00      0.000000        0          2         0  ioctl
 0.00      0.000000        0          1         1  faccessat
 0.00      0.000000        0          5         0  close
 0.00      0.000000        0          1         0  write
 0.00      0.000000        0          1         0  set_tid_address
 0.00      0.000000        0          1         0  futex
 0.00      0.000000        0          1         0  set_robust_list
 0.00      0.000000        0          2         0  rt_sigaction
 0.00      0.000000        0          3         0  rt_sigprocmask
 0.00      0.000000        0          1         0  uname
 0.00      0.000000        0          4         0  brk
 0.00      0.000000        0          1         0  clone
 0.00      0.000000        0          1         0  execve
 0.00      0.000000        0          6         0  mprotect
 0.00      0.000000        0          1         0  prlimit64
-----
100.00      0.002000          63         2 total
[root@localhost ~]#

```

c. Saída gerada pelo programa

```

luanna@DESKTOP-H7LHBMV: ~/Trabalho3
luanna@DESKTOP-H7LHBMV:~/Trabalho3$ gcc primos.c -o primos -lpthread
luanna@DESKTOP-H7LHBMV:~/Trabalho3$ ./primos
Insira o numero: 13
13 11 7 5 3 2

```