

NOME: Pedro Unello Neto

TIA: 41929713

A execuções abaixo são relacionadas ao exercício de aula da multiplicação matriz por vetor, aonde foi proposto à que se utilize de POSIX threads para paralelizar o grau mais alto dos dados na matriz (maior facilidade de reger integridade dos dados), de acordo com a função definida abaixo:

```
void *Pth_mat_vect(void* rank) {
    long my_rank = (long) rank;
    int i, j;
    int local_m = m/thread_count;
    int my_first_row = my_rank*local_m;
    int my_last_row = (my_rank+1)*local_m - 1;

    for (i = my_first_row; i <= my_last_row; i++) {
        y[i] = 0.0;
        for (j = 0; j < n; j++)
            y[i] += A[i][j]*x[j];
    }

    return NULL;
} /* Pth_mat_vect */
```

O exemplo que foi usado para a execução é dado na multiplicação da matriz e vetor abaixo.

Vetor x:

4
1
9
2
9
7

Matriz A:

6	4	2	5	3	1
1	6	4	2	5	3
3	1	6	4	2	5
5	3	1	6	4	2
2	5	3	1	6	4
4	2	5	3	1	6

Logo, a execução do código completo (disponibilizado neste mesmo repositório), utiliza 3 threads, para multiplicar um vetor de 6 posições por uma matriz 6 por 6, como pode ser visto abaixo:

Compilando com:

gcc -g -Wall -o MultMatrizVetorPthreads MultMatrizVetorPthreads.c -lpthread

```
pedrou@pop-os: ~/Documentos/VSCodeWS
pedrou@pop-os:~/Documentos/VSCodeWS$ gcc -g -Wall -o MultMatrizVetorPthreads MultMatrizVetorPthreads.c -lpthread
pedrou@pop-os:~/Documentos/VSCodeWS$ ./MultMatrizVetorPthreads
y = { 90.000000 , 116.000000 , 128.000000 , 94.000000 , 124.000000 , 120.000000 }
```

Da mesma forma, compilando com:

gcc -pthread -o MultMatrizVetorPthreads MultMatrizVetorPthreads.c

```
pedrou@pop-os: ~/Documentos/VSCodeWS
pedrou@pop-os:~/Documentos/VSCodeWS$ gcc -pthread -o MultMatrizVetorPthreads MultMatrizVetorPthreads.c
pedrou@pop-os:~/Documentos/VSCodeWS$ ./MultMatrizVetorPthreads
y = { 90.000000 , 116.000000 , 128.000000 , 94.000000 , 124.000000 , 120.000000 }
```

Por fim, alterando a função disponibilizada no exercício, para ilustrar melhor a ordenação e a execução propriamente ditam, como mostrado abaixo:

```
26 void *Pth_mat_vect( void* rank ) { //Função dado no exercício.
27     long my_rank = (long) rank;
28     int i, j;
29     int local_m = m/thread_count;
30     int my_first_row = my_rank*local_m;
31     int my_last_row = (my_rank+1)*local_m - 1;
32
33     //Alterações feitas apenas para ilustrar execução
34     printf("Thread %ld de %d\n", my_rank, thread_count);
35
36     for (i = my_first_row; i <= my_last_row; i++) {
37         y[i] = 0.0;
38
39         //Alterações feitas apenas para ilustrar execução
40         printf("y[%d] = ", i);
41
42         for (j = 0; j < n; j++)
43         {
44             y[i] += A[i][j]*x[j];
45             //Alterações feitas apenas para ilustrar execução
46             printf(" %i +", A[i][j]*x[j]);
47         }
48
49         //Alterações feitas apenas para ilustrar execução
50         printf("\n y[%d] = %f \n\n", i, y[i]);
51     }
52
53     return NULL;
54 } /* Pth_mat_vect */
55
56
```

Foram obtidos os seguintes resultados:

```
+ pedrou@pop-os: ~/Documentos/VSCodeWS
pedrou@pop-os:~/Documentos/VSCodeWS$ gcc -pthread -o MultMatrizVetorPthreads MultMatrizVetorPthreads.c
pedrou@pop-os:~/Documentos/VSCodeWS$ ./MultMatrizVetorPthreads
Thread 0 de 3
y[0] = 24 + 4 + 18 + 10 + 27 + 7 + 14 + 33 + 110
y[0] = 90.000000

y[1] = 4 + 6 + 36 + 4 + 45 + 21 + 12 + 54 + 18 + 35 + 20 + 3 + 9 + 12 + 36 + 14 + 8 + 5 + 27 + 2 + 54 + 28 + 16 + 2 + 45 + 6 + 9 + 42 + 10 + 27 + 7 + 14 + 33 + 110
y[1] = 116.000000

Thread 1 de 3
y[2] = 12 + 1 + 54 + 8 + 18 + 35 + 20 + 3 + 9 + 12 + 36 + 14 + 8 + 5 + 27 + 2 + 54 + 28 + 16 + 2 + 45 + 6 + 9 + 42 + 10 + 27 + 7 + 14 + 33 + 110
y[2] = 128.000000

y[3] = 20 + 3 + 9 + 12 + 36 + 14 + 8 + 5 + 27 + 2 + 54 + 28 + 16 + 2 + 45 + 6 + 9 + 42 + 10 + 27 + 7 + 14 + 33 + 110
y[3] = 94.000000

Thread 2 de 3
y[4] = 8 + 5 + 27 + 2 + 54 + 28 + 16 + 2 + 45 + 6 + 9 + 42 + 10 + 27 + 7 + 14 + 33 + 110
y[4] = 124.000000

y[5] = 16 + 2 + 45 + 6 + 9 + 42 + 10 + 27 + 7 + 14 + 33 + 110
y[5] = 120.000000

y = { 90.000000 , 116.000000 , 128.000000 , 94.000000 , 124.000000 , 120.000000 }
pedrou@pop-os:~/Documentos/VSCodeWS$
```

: