Aula 06 – OpenMP
Alunos: Luciana e Guillaume

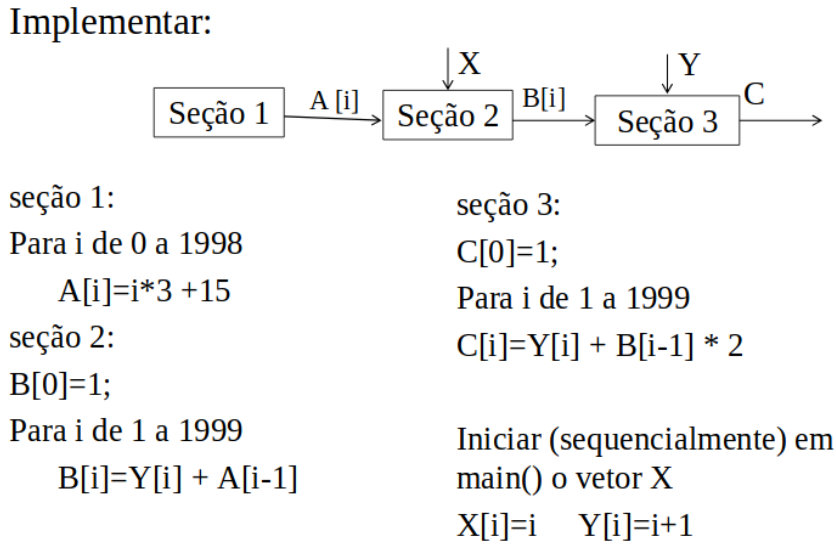
1. Tarefa 1

O objetivo da primeira tarefa era fazer um programa que faça R=(A x B) + (C x D), sendo as multiplicações AxB e C x D em uma região paralela cada, utilizando-se duas threads e #pragma parallel for. O código fonte está em tarefa1.c e o programa funciona conforme o esperado.

2. Tarefa 2

A segunda tarefa tem como objetivo implementar um programa conforme Figura 1.

Figura 1: Diagrama da Tarefa 2



A fim de se evitar que uma seção tentasse acessar uma casa ainda não preenchida pela seção anteior (por exemplo, se a seção 2 tentasse acessar uma posição de A ainda não inicializada), foram utilizados quatro semáforos.

Figura 2: Foram utilizados quatro semáforos

```
omp_lock_t sem1_doing;
omp_lock_t sem1_done;
omp_lock_t sem2_doing;
omp_lock_t sem2_done;
```

O primeiro semáforo é utilizado para que o processo A sinalize ao processo B que ainda está calculando o valor de A[i-1]. O segundo semáforo é utilizado para que o processo B sinalize que ainda está calculando seu valor para o processo A, e o terceiro é o mesmo, porém para sinalizar ao processo C. O último semáforo é utilizado pelo processo C para sinalizar que está efetuando seus cálculos.

O segundo e o quarto semáforo começam desbloqueados, pois presume-se que nenhum dos processos B e C ainda tinham iniciado.

Figura 3: Inicialização dos semáforos

```
omp_init_lock(&sem1_doing);
omp_init_lock(&sem1_done);
omp_init_lock(&sem2_doing);
omp_init_lock(&sem2_done);

omp_unset_lock(&sem1_done);
omp_unset_lock(&sem2_done);
```

Figura 4: Semáforos utilizado por A

```
#pragma omp parallel for shared(A) private(i)
for (i = 0; i < SIZE - 1; i++)
{
  omp_set_lock(&sem1_doing);
  A[i] = i*3 + 15;
  omp_unset_lock(&sem1_doing);
}
```

Figura 5: Semáforos utilizados por B

```
#pragma omp parallel for shared (B,Y,A) private(k)
for (k = 1; k < SIZE; k ++)
{
  omp_set_lock(&sem1_done);
  omp_set_lock(&sem2_doing);
  B[k] = Y[k] + A[k-1];
  omp_unset_lock(&sem2_doing);
  omp_unset_lock(&sem1_done);
}
```

Figura 6: Semáforos utilizados por C

```
#pragma omp parallel for shared(C,Y,B) private(x)
for (x = 1; x < SIZE; x++)
{
  omp_set_lock(&sem2_done);
  C[x] = Y[x] + B[x-1]*2;
  omp_unset_lock(&sem2_done);
}
```

Por fim, o programa imprime todos os valores das matrizes. No arquivo output.txt é possível verifica-los, sendo que este foi gerado pelo próprio terminal.

Figura 7: Execução pelo terminal

```
luciana@luciana-K46CA:~/Documents/Poli/PCS3868-Desempenho/aula06$ gcc tarefa2.c -o tarefa2 -fopenmp
luciana@luciana-K46CA:~/Documents/Poli/PCS3868-Desempenho/aula06$ ./tarefa2 > output.txt
```