



MODELO DE PARALELISMO DE DADOS

```

1. __kernel void dataParallel(__global float* A, __global float* B, __global
float* C)
2. {
3. int base = 4*get_global_id(0);
4. C[base+0] = A[base+0] + B[base+0];
5. C[base+1] = A[base+1] - B[base+1];
6. C[base+2] = A[base+2] * B[base+2];
7. C[base+3] = A[base+3] / B[base+3];
8. }

```

```
51. /* Initialize input data */
52. for (i=0; i < 4; i++) {
53.     for (j=0; j < 4; j++) {
54.         A[i*4+j] = i*4+j+1;
55.         B[i*4+j] = j*4+i+1;
56.     }
57. }
```

MODELO DE PARALELISMO DE TAREFA

```
1. __kernel void taskParallelAdd(__global float* A, __global float* B, __global float* C)
2. {
3.     int base = 0;
4.
5.     C[base+0] = A[base+0] + B[base+0];
6.     C[base+4] = A[base+4] + B[base+4];
7.     C[base+8] = A[base+8] + B[base+8];
8.     C[base+12] = A[base+12] + B[base+12];
9. }
```

```
11. __kernel void taskParallelSub(__global float* A, __global float* B, __gl
obal float* C)
12. {
13.     int base = 1;
14.
15.     C[base+0] = A[base+0] - B[base+0];
16.     C[base+4] = A[base+4] - B[base+4];
17.     C[base+8] = A[base+8] - B[base+8];
18.     C[base+12] = A[base+12] - B[base+12];
19. }
```

```
21. __kernel void taskParallelMul(__global float* A, __global float* B, __gl  
obal float* C)
```

```
22. {
```

```
23. int base = 2;
```

```
24.
```

```
25. C[base+0] = A[base+0] * B[base+0];
```

```
26. C[base+4] = A[base+4] * B[base+4];
```

```
27. C[base+8] = A[base+8] * B[base+8];
```

```
28. C[base+12] = A[base+12] * B[base+12];
```

```
29. }
```

```
31. __kernel void taskParallelDiv(__global float* A, __global float* B, __gl  
obal float* C)
```

```
32. {
```

```
33. int base = 3;
```

```
34.
```

```
35. C[base+0] = A[base+0] / B[base+0];
```

```
36. C[base+4] = A[base+4] / B[base+4];
```

```
37. C[base+8] = A[base+8] / B[base+8];
```

```
38. C[base+12] = A[base+12] / B[base+12];
```

```
39. }
```

```
52. /* Initialize input data */
```

```
53. for (i=0; i < 4; i++) {
```

```
54.     for (j=0; j < 4; j++) {
```

```
55.         A[i*4+j] = i*4+j+1;
```

```
56.         B[i*4+j] = j*4+i+1;
```

```
57.     }
```

```
58. }
```