

**Començat el** diumenge, 19 de desembre 2021, 19:01

**Estat** Acabat

**Completat el** diumenge, 19 de desembre 2021, 19:17

**Temps emprat** 16 minuts 27 segons

**Qualificació** 7,00 sobre 7,00 (100%)

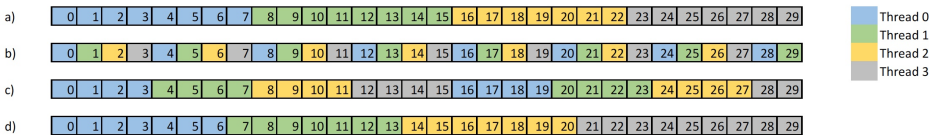
Pregunta 1  
Correcte  
Puntuació 1,00 sobre 1,00

Given the following parallel region in OpenMP initializing vector a:

```
#define n 30
int vector[n];

#pragma omp parallel num_threads(4)
{
    int myid = omp_get_thread_num( );
    int howmany = omp_get_num_threads( );
    int num_elems = n / howmany;
    int start = myid * num_elems;
    int end = start + num_elems;
    if ( myid == howmany-1 ) end = n;
    for ( i = start; i < end; i++ ) {
        vector[i] = foo(i);
    }
}
```

Indicate which one of the following OUTPUT data decompositions for vector is being applied to drive the parallelization of the code using implicit tasks:



**Note:** a) corresponds to a block data decomposition in which threads 0 and 1 have 8 consecutive elements and threads 2 and 3 only 7; b) corresponds to a cyclic data decomposition, with the first element assigned to thread 0; c) corresponds to a block-cyclic data decomposition with block size equal to 4; and finally, d) corresponds to a block data decomposition in which threads 0, 1 and 2 have 7 consecutive elements each and thread 3 has 9.

- Trieu-ne una:
- ☐ Data decomposition a)
  - ☐ Data decomposition b)
  - ☐ Data decomposition c)
  - ☒ Data decomposition d)

✔ Well done!

La teva resposta és correcta.

Pregunta 2  
Correcte  
Puntuació 1,00 sobre 1,00

If in the previous code we change the line:

```
if ( myid == howmany-1 ) end = n;
```

to the following code sequence:

```
int rem_elems = n % howmany;
if (rem_elems != 0) {
    if (myid < rem_elems) {
        start += myid;
        end += (myid+1);
    } else {
        start += rem_elems;
        end += rem_elems;
    }
}
```

indicate which one of the OUTPUT data decompositions shown in the first question would be applied to drive the parallelization of the code using implicit tasks:

Trieu-ne una:

- ☒ Data decomposition a)
- ☐ Data decomposition b)
- ☐ Data decomposition c)
- ☐ Data decomposition d)

✓ Well done!

La teva resposta és correcta.

Pregunta **3**

Correcte

Puntuació 1,00 sobre 1,00

If the code is changed to:

```
#pragma omp parallel
{
    int myid = omp_get_thread_num( );
    int howmany = omp_get_num_threads( );
    for (int i = myid; i < n; i+=howmany) {
        vector[i] = foo(i);
    }
}
```

Indicate which one of the OUTPUT data decompositions shown in the first question is being applied to drive the parallelization of the code using implicit tasks:

Trieu-ne una:

- ☐ Data decomposition a)
- ☒ Data decomposition b)
- ☐ Data decomposition c)
- ☐ Data decomposition d)

✓ Well done!

La teva resposta és correcta.

Pregunta **4**

Correcte

Puntuació 1,00 sobre 1,00

How would you change the loop in the question 3 in order to follow the data decomposition that has not been applied in any of the previous three codes? You can assume that constant BS is defined as 4.

Trieu-ne una:

- ☐ I just simply need to change the i loop to go from myid\*BS to n in steps of 1, as follows:

```
for (int i = myid*BS; i < n; i++)
{ ... }
```

- ☐ I just simply need to change the i loop to go from myid\*BS to n in steps of BS, as follows:

```
for (int i = myid*BS; i < n; i += BS)
{ ... }
```

- ☐ I just simply need to change the i loop to go from myid\*BS to n in steps of howmany\*BS, as follows:

```
for (int i = myid*BS; i < n; i += howmany*BS)
{ ... }
```

- ☒ I need to add a new loop to the previous code traversing the consecutive elements inside each block, as follows:

```
for (int ii = myid*BS; ii < n; ii += howmany*BS)
for (int i = ii; i < min(ii+BS, n); i++)
{ ... }
```

Correct! You need two loops, one to traverse the blocks assigned to each implicit task and a second one to traverse the iterations inside each block.

La teva resposta és correcta.

Pregunta **5**

Correcte

Puntuació 1,00 sobre 1,00

Of the four proposed data decompositions for vector shown in the first question, and assuming that the execution time for function foo is constant, only data decompositions labeled a) and b) may introduce a load unbalance (measured in terms of the difference in the number of elements assigned to implicit tasks) of 1 data element. Data decomposition c) may have a maximum load unbalance (i.e. the largest difference between number of elements assigned to two different implicit tasks) equal to the size of the block while data decomposition d) a maximum load unbalance equal to the number of processors used minus 1.

Trieu-ne una:

- ☒ Vertader ✓
- ☐ Fals

Well done!

Pregunta **6**

Correcte

Puntuació 1,00 sobre 1,00

If the parallel region is changed as follows in order to do a certain computation with the elements of vector:

```
#pragma omp parallel reduction(+:sum)
{
    int myid = omp_get_thread_num( );
    int howmany = omp_get_num_threads( );
    for (int i = 0; i < n; i++) {
        int j = swap(i, n);
        if (j%howmany) == myid) sum += foo(vector[j], i);
    }
}
```

```
}
```

being swap a function that returns an integer in the range 0 to n-1 based on the value of i. Indicate which one of the INPUT data decompositions for vector shown in the first question is being applied to drive the parallelization of the code using implicit tasks:

Trieu-ne una:

- ☒ INPUT data decomposition b)
- ☐ OUTPUT data decomposition b)
- ☐ INPUT data decomposition c)
- ☐ OUTPUT data decomposition c)

✓ Well done!

La teva resposta és correcta.

Pregunta 7

Correcte

Puntuació 1,00 sobre 1,00

Finally, if the previous code is changed as follows:

```
#pragma omp parallel reduction(+:sum)
{
    int myid = omp_get_thread_num( );
    int howmany = omp_get_num_threads( );
    int num_elems = n / howmany;
    int start = myid * num_elems;
    int end = start + num_elems;
    if (myid == howmany-1) end = n;
    for (int i = 0; i < n; i++) {
        int j = swap(i, n);
        if ((j>=start) && (j<end)) sum += foo(a[j], i);
    }
}
```

Indicate which one of the INPUT data decompositions for vector a shown in the first question is being applied to drive the parallelization of the code using implicit tasks:

Trieu-ne una:

- ☐ INPUT data decomposition a)
- ☐ OUTPUT data decomposition a)
- ☒ INPUT data decomposition d)
- ☐ OUTPUT data decomposition d)

✓ Well done!

La teva resposta és correcta.

◀ Slides for Unit 5: data-aware task decomposition strategies

Salta a...

Lab1 laboratory assignment ▶