

CompSci-230: Homework 3

Due 11-February-2019, in pdf format on CANVAS.

Consider the following three-level nested loop program to multiply two matrices A and B on a single processor system:

```
For  $i = 0, n - 1$ , do:
  For  $j = 0, n - 1$ , do:
    For  $k = 0, n - 1$ , do:
       $c_{i,j} = c_{i,j} + a_{ik} \times b_{k,j}$ 
    endfor
  endfor
endfor
```

This is the exact same nested loop program as discussed in class and available in the class' slides. We discussed three different ways of performing this computation on a n -processor ring when unfolding the outermost loop with the index i .

It is well known that changing the order of the loops is a program transformation that does not alter the final result (an invariant transformation.) The three nested loops can appear in one out of 6 permuted possibilities, namely: (i, j, k) , (i, k, j) , (j, i, k) , (j, k, i) , (k, i, j) , (k, j, i) .

1. Assuming execution of the above program in a single processor computer, characterize the manner in which the computation progresses for each one of the 6 possible nested loop cases: provide a data to memory mapping, and show how the main arithmetic expression progresses, i.e. how the inner products are computed in time, as the indexes advance with the innermost loop being the one that advances fastest.
2. In a manner akin to the one used in class (see slides), for each one of the nested loop cases, discuss the mapping of the computation on a n -processor ring if, for each of the 6 cases, the outermost loop is used to unfold and parallelize the computation.