

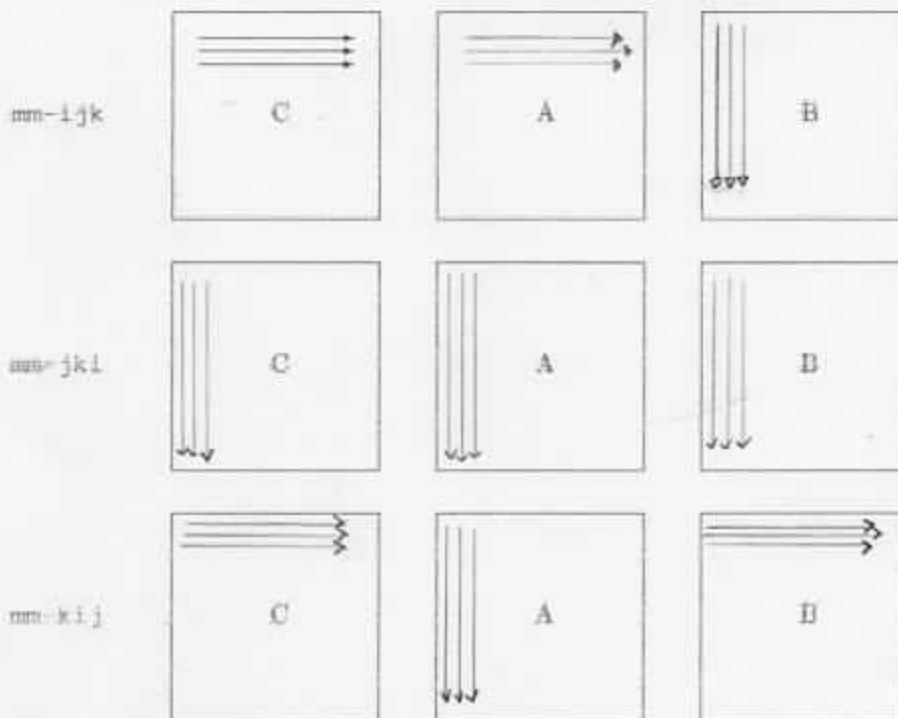
Nombre: LEO ARRIOLA HEIKUE

Grupo: A

Nombre: ERIC SERRANO

### Hoja de respuesta al Estudio Previo

1. Dibujad, para cada una de las formas  $ijk$ , en qué orden se recorren las matrices A, B y C:



2. Calculad los fallos de cache, suponiendo que la MC es de tamaño infinito y completamente asociativa. Escribid los resultados en la siguiente tabla:

N	mm-ijk			mm-jki			mm-kij		
	matriz A	matriz B	matriz C	matriz A	matriz B	matriz C	matriz A	matriz B	matriz C
256	32	256	1	256	1	256	1	32	32
512	64	512	1	512	1	512	1	64	64
1024	128	1024	1	1024	1	1024	1	128	128

3. Calculad cuántas páginas de memoria virtual se utilizan al ejecutar completamente el bucle más interno 1 vez. Escribid los resultados en la siguiente tabla:

N	mm-ijk			mm-jki			mm-kij		
	matriz A	matriz B	matriz C	matriz A	matriz B	matriz C	matriz A	matriz B	matriz C
256	1	32	1	32	1	32	1	1	1
512	1	128	1	128	1	128	1	1	1
1024	1	512	1	512	1	512	1	1	1

Nombre: LEO ARRIOLA MEIKLE

Grupo: A

Nombre: ERIC SERRANO

## Hoja de respuestas de la práctica

1. Compilad y ejecutad los tres programas para un tamaño  $N=6$ . Comprobad que los 3 programas dan el mismo resultado.
2. Rellenad la siguiente tabla:

N	Tiempo ejecución (en seg.)			MFLOPS		
	mm-ijk	mm-jki	mm-kij	mm-ijk	mm-jki	mm-kij
256	0,105s	0,101s	0,084s	349,5	332,2	391,4
512	0,667	0,743	0,422	402,4	361,28	628,65
1024	47,91s	22,93s	3,43s	109,81	49,65	550,

3. Teniendo en cuenta lo que habéis hecho en los apartados anteriores y en el trabajo previo, explicad la razón de las diferencias de rendimiento en estos tres programas.

Podemos ver de nuevo como a medida que aumenta  $N$  el tiempo de ejecución también lo hace, esto se debe al mayor número de folios. También, respecto al acceso de los matrices, podemos ver de nuevo como el acceso por fila aumenta el tiempo considerablemente.

4. Apliqué la optimización adicional a las otras dos aplicaciones. Compilad y ejecutad los tres programas para un tamaño  $N=6$ . Comprobad que los 3 programas dan el mismo resultado.
5. Rellenad la siguiente tabla:

N	Tiempo ejecución (en seg.)			MFLOPS		
	mm-ijk	mm-jki	mm-kij	mm-ijk	mm-jki	mm-kij
256	0,054	0,057	0,046	621,3	588,67	729,44
512	0,383	0,4	0,322	427,4	671,09	833,55
1024	6,34	7,1	2,55	338,71	302,46	842,45

6. Comparad los resultados obtenidos con los obtenidos antes de optimizar los programas, y sacad conclusiones de dicha comparación.

Podemos apreciar que ha habido un gran ahorro considerable, esto se debe a que al tener el vector de la matriz  $C$  guardado con anterioridad en una variable, evitamos los posibles fallos de acceso a este.