

Inicializar MPI

Si (myrank == 0)

 leer el número de vértices en el grafo (n)

Hacer un Broadcast de n

Si (myrank == 0)

 Dimensionar la matriz de pesos, *dist*: Crear_matriz_pesos_consecutivo(n,n)

 Dimensionar la matriz de caminos, *caminos*: Crear_matriz_caminos_consecutivo(n,n)

 Crear el grafo: Definir_Grafo(...)

 Calcular el reparto de filas entre procesos

$n_{filas} = n / numprocs$

$resto = n \% numprocs$

$n_{local} = n_{filas} + resto$

else

 Calcular el reparto de filas

$n_{filas} = n / numprocs$

$resto = n \% numprocs$

$n_{local} = n_{filas}$

 Dimensionar la matriz de pesos: Crear_matriz_pesos_consecutivo(nlocal,n)

 Dimensionar la matriz de caminos: Crear_matriz_caminos_consecutivo(nlocal,n)

ALGORITMO DE FLOYD-WARSHALL

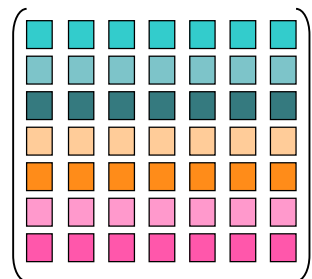
(esquema)

```
Utilizar MPI_Scatter para repartir las matrices de peso y de caminos
Reservar memoria para aux y auxC, float y entero de tamaño n, respect.
Empezar las iteraciones:
Para (k=0; k<n; k++)
    Calcular el número de proceso que almacena la fila k: sender
    Si (myrank == sender)
        Hacer aux igual a la fila k de la matriz de pesos, dist.
        Hacer auxC igual a la fila k de la matriz de caminos, caminos.
    Hacer un Broadcast de aux y auxC
    Actualizar matrices:
    Para (i = 0; i < nlocal; ++i)
        Para (j = 0; j < n; ++j)
            Si ((dist[i][k] * aux[j] != 0) )
                Si ((dist[i][k] + aux[j] < dist[i][j]) || (dist[i][j] == 0))
                    dist[i][j] = dist[i][k] + aux[j];
                    caminos[i][j]=auxC[j];
Usar MPI_Gather para recoger las matrices de peso y de caminos
Si (myrank == 0) calcula_camino(dist, caminos, n);
```

ALGORITMO DE FLOYD-WARSHALL

(esquema)

Calcular el número de proceso que almacena la fila k: *sender*



Iteración	sender	nfilas	resto	i	resto+nfilas*(i+1)
k=0,1,2	0	2	1	0,1,2	3,5,7
k=3,4	1	2	1		
k=5,6	2	2	1		

```

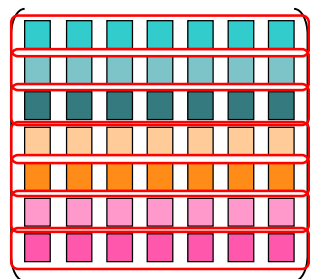
Para (i=0; i<numprocs; i++){
  Si (k<resto+nfilas*(i+1))
    sender = i;
    break;

```

ALGORITMO DE FLOYD-WARSHALL

(esquema)

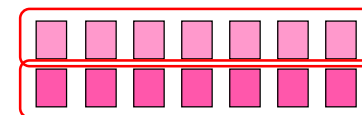
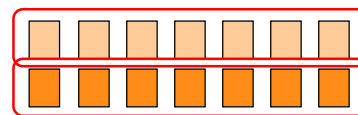
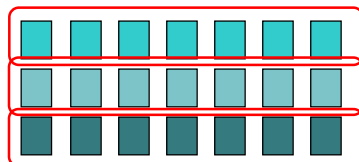
¿En qué fila de mis arrays *dist* y *caminos* está la fila *k* que busco?



P0

P1

P2



Filas antes:

$k - \text{filas antes}$

P0: 0

0

1

2

P1: 3

0

1

P2: 5

0

1

$k = 0$: la fila 0 está en la **posición 0** del proceso 0

$k = 1$: la fila 1 está en la **posición 1** del proceso 0

$k = 2$: la fila 2 está en la **posición 2** del proceso 0

$k = 3$: la fila 3 está en la **posición 0** del proceso 1

$k = 4$: la fila 4 está en la **posición 1** del proceso 1

$k = 5$: la fila 5 está en la **posición 0** del proceso 2

$k = 6$: la fila 6 está en la **posición 1** del proceso 2