

Filósofos

1. El filósofo solo se tiene que preocupar de pedir primero el tenedor izquierdo y después de pedir el derecho, comer y soltarlos, por tanto, eso es lo que debe hacer el filósofo.

Primero se hace la petición al tenedor de su izquierda, como procese el tenedor esa petición es transparente para él.

Se le envía la etiqueta COMER para que el tenedor sepa qué solicita el tenedor.

Después se hace la petición al tenedor de su derecha y ocurre igual que con el de la izquierda.

Para soltar los tenedores se hace la misma operación. Se le envían dos peticiones, una a cada tenedor, con la etiqueta PENSAR para avisar que el tenedor se va a soltar.

—

En el proceso Tenedor no es necesario saber de donde viene la petición, solo con saber su etiqueta, el proceso se encargará, si está disponible, de seguir su ejecución.

Este proceso no hace otra cosa más que esperar peticiones para poder continuar su ejecución. Si hay dos procesos que solicitan el mismo tenedor, solo uno de ellos será respondido y hasta que el filósofo no lo suelte no podrá recibir otra.

—

Si todos los filósofos cogen su tenedor a la izquierda a la vez es cuando se produce el interbloqueo, ya que no habría ningún tenedor a la derecha disponible y todos estarían esperando que el tenedor recibiese su petición. A la vez, el tenedor estaría esperando que el filósofo le avisase que ha terminado de comer, pero como ni han empezado, nunca recibirían esa petición.

—

La solución es que uno de ellos coja el tenedor a su derecha en primer lugar. Da igual que filósofo sea, pero que uno de ellos la coja al contrario del resto, así, si coge primer el tenedor de la derecha, tendrá su tenedor de la izquierda siempre libre y si no coge el tenedor de la derecha porque ya ha sido cogido, tendrá que esperar a que se libere, dejando su tenedor de la izquierda libre para que otro proceso pueda coger el de su derecha.

2. Código fuente completo:

```
////////////////////////////////////  
//  
// Francisco Javier Caracuel Beltrán  
//  
// 2º B - Grupo B3  
//  
// Grado en Ingeniería Informática - 2015/2016 - SCD  
//  
// Práctica 3. Filósofos  
//  
////////////////////////////////////
```

```

#include <iostream>
#include <time.h>    // incluye "time"
#include <unistd.h>  // incluye "usleep"
#include <stdlib.h>  // incluye "rand" y "srand"
#include <mpi.h>

#define COMER 1
#define PENSAR 2

using namespace std;

void Filosofo(int id, int nprocesos);
void Tenedor(int id, int nprocesos);

////////////////////////////////////
// main
//
int main(int argc, char** argv) {

    int rank, size;

    srand(time(0));

    MPI_Init(&argc, &argv);

    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &size);

    if (size != 10) {

        if (rank == 0)
            cout << "El numero de procesos debe ser 10" << endl << flush;

        MPI_Finalize();

        return 0;

    }

    if ((rank % 2) == 0)
        Filosofo(rank, size); // Los pares son Filsofos
    else
        Tenedor(rank, size); // Los impares son Tenedores

    MPI_Finalize();

    return 0;

}

```

```

////////////////////////////////////
// Filósofo
//
void Filosofo(int id, int nprocesos) {

    int izq = (id + 1) % nprocesos;
    int der = ((id + nprocesos) - 1) % nprocesos;

    while (1) {

        switch(id){

            case 0:

                // Solicita tenedor derecho
                cout << "Filosofo " << id << " coge tenedor der ..." << der << endl << flush;
                MPI_Ssend(&der, 1, MPI_INT, der, COMER, MPI_COMM_WORLD);

                // Solicita tenedor izquierdo
                cout << "Filosofo " << id << " solicita tenedor izq ..." << izq << endl << flush;
                MPI_Ssend(&izq, 1, MPI_INT, izq, COMER, MPI_COMM_WORLD);

                break;

            default:

                // Solicita tenedor izquierdo
                cout << "Filosofo " << id << " solicita tenedor izq ..." << izq << endl << flush;
                MPI_Ssend(&izq, 1, MPI_INT, izq, COMER, MPI_COMM_WORLD);

                // Solicita tenedor derecho
                cout << "Filosofo " << id << " coge tenedor der ..." << der << endl << flush;
                MPI_Ssend(&der, 1, MPI_INT, der, COMER, MPI_COMM_WORLD);

        }

        cout << "Filosofo " << id << " COMIENDO" << endl << flush;
        sleep((rand() % 3) + 1); //comiendo

        // Suelta el tenedor izquierdo
        cout << "Filosofo " << id << " suelta tenedor izq ..." << izq << endl << flush;
        MPI_Ssend(&izq, 1, MPI_INT, izq, PENSAR, MPI_COMM_WORLD);

        // Suelta el tenedor derecho
        cout << "Filosofo " << id << " suelta tenedor der ..." << der << endl << flush;
        MPI_Ssend(&der, 1, MPI_INT, der, PENSAR, MPI_COMM_WORLD);

        // Piensa (espera bloqueada aleatorio del proceso)
        cout << "Filosofo " << id << " PENSANDO" << endl << flush;
    }
}

```

```

        // espera bloqueado durante un intervalo de tiempo aleatorio
        // (entre una décima de segundo y un segundo)
        usleep(1000U * (100U + (rand() % 900U)));

    }

}

////////////////////////////////////
// Tenedor
//
void Tenedor(int id, int nprocesos) {

    int buf;
    MPI_Status status;
    int filo;

    while (true) {

        // Espera una petición desde cualquier filósofo vecino ...
        MPI_Recv(&buf, 1, MPI_INT, MPI_ANY_SOURCE, COMER, MPI_COMM_WORLD,
        &status);

        // Recibe la petición del filósofo ...
        filo = status.MPI_SOURCE;

        cout << "Ten. " << id << " recibe petic. de " << filo << endl << flush;

        // Espera a que el filósofo suelte el tenedor...
        MPI_Recv(&buf, 1, MPI_INT, filo, PENSAR, MPI_COMM_WORLD, &status);
        cout << "Ten. " << id << " recibe liberac. de " << filo << endl << flush;

    }

}

```

3. Listado parcial de la salida:

```

fran@fran-Lenovo-Ubuntu:~/Escritorio/Universidad/SCD/Mis prácticas/Práctica 3/Filósofos$
mpicxx -o filosofos filosofos.cpp
fran@fran-Lenovo-Ubuntu:~/Escritorio/Universidad/SCD/Mis prácticas/Práctica 3/Filósofos$
mpirun -np 10 ./filosofos
Filosofo 4 solicita tenedor izq ...5
Filosofo Ten. 5 recibe petic. de 4
4 coge tenedor der ...3
Filosofo 0 coge tenedor der ...9
Filosofo Filosofo 8 solicita tenedor izq ...9
2 solicita tenedor izq ...3
Ten. 3 recibe petic. de 2
Filosofo Ten. 9 recibe petic. de 0

```

6 solicita tenedor izq ...7
Filosofo Ten. 7 recibe petic. de 6
6 coge tenedor der ...5
Filosofo 0 solicita tenedor izq ...1
Filosofo 2 coge tenedor der ...1
Ten. 1 recibe petic. de 0
Filosofo 0 COMIENDO
Filosofo 0 suelta tenedor izq ...1
Ten. 1 recibe liberac. de 0
Ten. 1 recibe petic. de 2
Filosofo 2 COMIENDO
Filosofo 0 suelta tenedor der ...9
Ten. 9 recibe liberac. de 0
Ten. Filosofo 8 coge tenedor der ...7
9 recibe petic. de 8
Filosofo 0 PENSANDO
Filosofo 0 coge tenedor der ...9
Filosofo 2 suelta tenedor izq ...3
Ten. 3 recibe liberac. de 2
Ten. 3 recibe petic. de 4
Filosofo 2 suelta tenedor der ...1
Ten. 1 recibe liberac. de 2
Filosofo 2 PENSANDO
Filosofo 4 COMIENDO
Filosofo 2 solicita tenedor izq ...3
Filosofo 4 suelta tenedor izq ...5
Filosofo 4 suelta tenedor der ...3
Ten. 5 recibe liberac. de 4
Ten. 5 recibe petic. de 6
Filosofo 6 COMIENDO
Ten. 3 recibe liberac. de 4Filosofo 2 coge tenedor der ...1
Filosofo 2 COMIENDO

Ten. 3 recibe petic. de 2
Filosofo Ten. 1 recibe petic. de 2
4 PENSANDO
Filosofo 4 solicita tenedor izq ...5
Ten. 5 recibe liberac. de 6
Ten. 5 recibe petic. de 4
Filosofo 6 suelta tenedor izq ...7
Filosofo 6 suelta tenedor der ...5
Filosofo 6 PENSANDO
Ten. 7 recibe liberac. de 6
Ten. 7 recibe petic. de 8
Filosofo 8 COMIENDO
Filosofo 4 coge tenedor der ...3
Filosofo 6 solicita tenedor izq ...7
Filosofo 8 suelta tenedor izq ...9
Filosofo Ten. 9 recibe liberac. de 8
Ten. 9 recibe petic. de 0
Filosofo 0 solicita tenedor izq ...1
8 suelta tenedor der ...7

Ten. 7 recibe liberac. de 8
Filosofo Filosofo 6 coge tenedor der ...5
Ten. 7 recibe petic. de 6
8 PENSANDO
Filosofo 8 solicita tenedor izq ...9
Filosofo 2 suelta tenedor izq ...3
Ten. 3 recibe liberac. de 2
Ten. 3 recibe petic. de 4
Filosofo 2 suelta tenedor der ...1
Filosofo 4 COMIENDO
Ten. 1 recibe liberac. de 2
Ten. 1 recibe petic. de 0
Filosofo 2 PENSANDO
Filosofo 0 COMIENDO
Filosofo 2 solicita tenedor izq ...3
Filosofo 4 suelta tenedor izq ...5
Ten. 5 recibe liberac. de 4
Ten. 5 recibe petic. de 6
Filosofo 0 suelta tenedor izq ...1
Filosofo 4 suelta tenedor der ...3
Filosofo 6 COMIENDO
Ten. 3 recibe liberac. de 4
Ten. 3 recibe petic. de 2
Ten. 1 recibe liberac. de 0
Filosofo 0 suelta tenedor der ...9
Filosofo 2 coge tenedor der ...1
Filosofo 4 PENSANDO
Ten. 9 recibe liberac. de 0
Ten. 9 recibe petic. de 8
Ten. 1 recibe petic. de 2
Filosofo 8 coge tenedor der ...7
Filosofo 2 COMIENDO
Filosofo 0 PENSANDO