Exercise 3

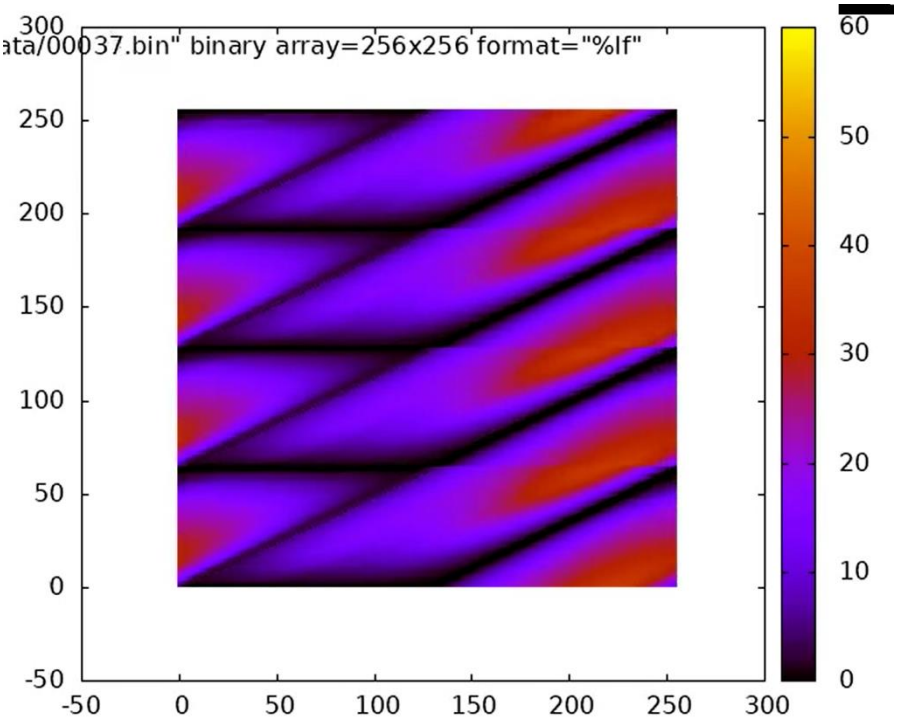
Louka CHAPUS

Before doing anything, we need to determine the size of each dimension to have the right number for the communicator. So, we need to decompose the number of sizes into integer divisor, and we add all integer divisor in a list. The size of each dimension is the number in the middle of the list of integer divisor.

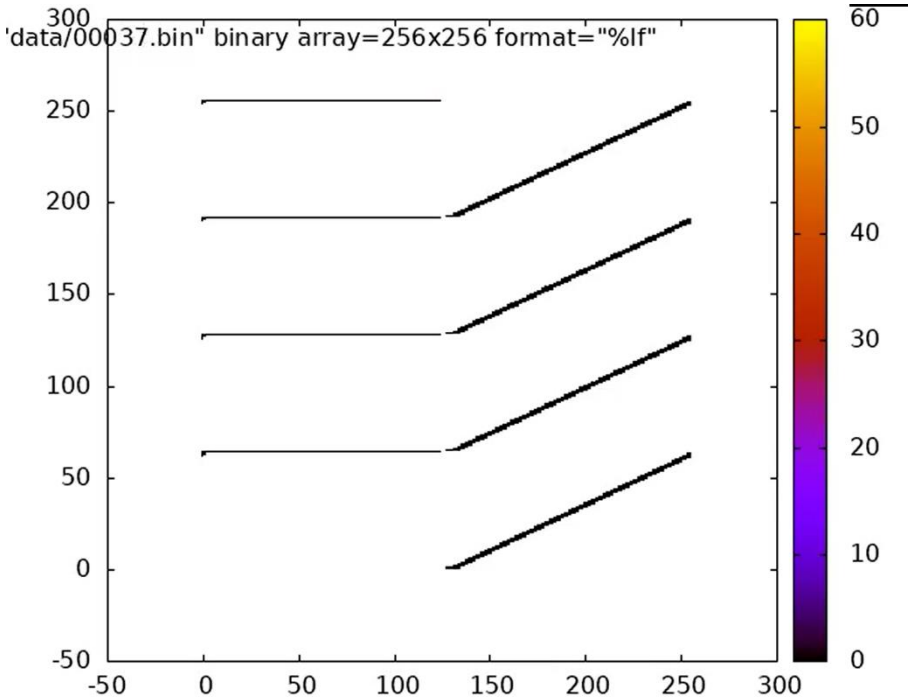
```
int *prime_number;
int lengh = 0;
prime_number = (int *)malloc(lengh * sizeof(int));      // make a dynamic array

for(int k=1; k<= size; k++){
    if(size%k == 0){
        lengh++;
        prime_number = (int *)realloc(prime_number, lengh * sizeof(int)); // increase le lengh of the array
        prime_number[lengh-1] = k;                                        // add the value
    }

}
if(lengh%2 == 0){                                   // if the lengh is even
    dim_size[0] = prime_number[(lengh/2) - 1];      // we take the two number close
    dim_size[1] = prime_number[lengh/2];            // to the center
}
else{                                               // else the lengh is odd
    dim_size[0] = prime_number[lengh/2];            // we take the element in the center
    dim_size[1] = prime_number[lengh/2];
}

free(prime_number);
```

Then we can continue with all the MPI stuff, we create the cartesian communicator without periodic boundaries. And we parse argument to the rank 0 and broadcast to other process. Then there are some problems, I manage to implement the cartesian communicator and get it work. I can also use the Cart_Shift to see the neighbour of each cell (it's -2 when there is no neighbour). I have also the time_step, initialization and finalize who work well. But my main problem where to save the domain, I tried a lot of different method but none of them print the right picture only the function for the exercise 2 and with one process can print the right video. Next, is the different result depending on with function is enable. And also the program crash at the end of the execution but we can still do the video.

Without boundaries condition, without boundaries exchange



With boundaries condition, without boundaries exchange



With boundaries exchange, without boundaries conditions