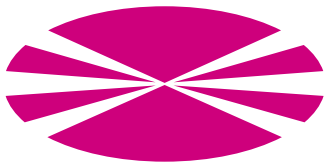
Concurrency and Parallelism. Block II Parallelism

Assignment 1: count the number of occurrences of a letter

Spring 2023



Count the number of occurrences of a letter

Sequential code

```
int i, n, count=0;
char *cadena;
char L;

n = atoi(argv[1]);
L = *argv[2];

cadena = (char *) malloc(n*sizeof(char));
inicializaCadena(cadena, n);

for(i=0; i<n; i++){
  if(cadena[i] == L){
    count++;
  }
}

printf("El numero de apariciones de la letra %c es %d\n", L, count);
free(cadena);
```

Count the number of occurrences of a letter

Parallelization

- SPMD implementation
- I/O (argv[]/printf) is made by process 0
- Distribute n and L to all the processes (with Send/Recv)
- Divide the workload of the for loop with "step" i+=numprocs instead of i++
- Gather the number of occurrences detected by each process (with Send/Recv)

Conditions of the assignment

- Value: 0.25 points
- Deadline: 17-21 April
- Must be done in couples and defended in the laboratory class