Assignment 2

Take the general problems, which have been discussed in the lab classes and for which you have developed a multithreaded solution. The aim now is to convert them into a multiprocess message passing application using the MPI library and running under Linux.

The decomposition in both cases can be described by the following diagram.

So, the role of the process *dispatcher* will be getting the data file names by processing the command line, distributing to the *worker* processes chunks of data to be processed, waiting for their processing and saving partial results, and, when all work is done, let them know of the fact and print the results of the whole processing. On the other hand, the role of the processes *worker* will be carrying out the processing itself: they request in succession pieces of data to process, process it and deliver the results of their processing. They terminate when there are no more data pieces to process.

GRADING

- development and validation of a multiprocess message passing application using the MPI library of one of the general problems according to specification – 13 points
- development and validation of a multiprocess message passing application using the MPI library of the other general problem according to specification – 20 points.

DELIVERABLES

- an archive, named `CLE2_T$G#.zip` (where `$`, equal to 1, ... , 3, means the lab number, and `#`, equal to 1, ... , 10, means the group number) containing both the source files of your solution to the two problems and a pdf file, named `present.pdf`, up to 6 pages (power point like), where the main ideas of the design of the solutions to the two problems and the timing results that were obtained on runs with 1, 2, 4, 8 *worker* processes, are discussed
- the archive should result from the compression of a directory, named `CLE2_T$G#`, containing two subdirectories, named `prog1` and `prog2`, and the file `present.pdf`.

DEADLINE

May, 29, at midnight.