

Versiunea paralela

-MPI-

~Proiect APD~

Safta Maria-Andreea

Tema proiect : Text Processing

Pentru realizarea versiunii paralele a acestei tema m-am folosit de functia de tokenizare , normalizare a unui text si de metoda cu procese multiple , MPI, iar ca limbaj de programare am folosit C++.

Programul incepe cu initializarea mediul MPI apoi se incepe masurarea timpul de inceput al executiei pentru a calcula durata totala pentru final .

Pentru citirea fisierului se va folosi “buffer << file.rdbuf();” ,apoi cu “string text = buffer.str();” se va incepe convertirea continutului intr-un string . Cuvintele se vor imparti fiecarui proces , iar in “localWordCounts” se vor stoca frecventele cuvintelor. Procesul cu rank 0 va colecta si va afisa rezultatele in ordinea alfabetica . “MPI_Send” trimite numarul de intrari , lungimea cuvantului si frecventa sa catre rank 0 . La sfarsit de va genera timpul total de milisecunde necesar pentru realizarea nuamrarii frecventelor cat si ordonarea acestora .

Pentru a putea rula programul realizat cu MPI , vom scrie in cmd mpiexec -n nr procese NumeProiect input.txt

Date experimentale :

```
(C) Microsoft Corporation. All rights reserved.  
C:\Users\Andreea\Desktop\AN3_sem2\APD\ProiectAPD\x64\Debug>mpiexec -n 2 ProiectAPD input.txt  
Frecventa cuvantului "Ballets": 1  
Frecventa cuvantului "Giselle": 1  
Frecventa cuvantului "L'apres-midi": 1  
Frecventa cuvantului "La": 1  
Frecventa cuvantului "Le": 1  
Frecventa cuvantului "Antrenamentele": 1  
Frecventa cuvantului "Astazi": 1  
Timpul total de rulare: 15 milisecunde
```

Pentru ca am folosit MPI si putem sa rulam programul pe mai multe procese pentru testare am ajuns la concluzia ca indiferent de numarul proceselor alese timpul a iest aproximativ de 15 ms.