# Multiprocesorski sistemi – Zadatak 1 – Merge Sort

Ovaj document ukratko opisuje nacin rada algoritma za Merge Sort. Koriscen je kod preuzet sa interneta za sam algoritam Merge sortiranja I time je “kreiran” sekvencijalni program uz dodato merenje vremena I randomizovani niz.

Za paralelnu verziju koda koriscena je ista sama funkcija MergeSort kao iz sekvencijalnog koda(single core koda), kreiran je randomizovani niz koji se upisuje u datoteku input.txt. Pokrenuto je merenje vremena. Zatim se ulazi u pragmu gde se preko argumenta zadaje tc tj. Broj niti koji ce raditi unutar pragme. Blokovskom raspodelom je realizovana podela pocetnog nita izmedju niti, za sve slucajeve (2,3,4…), a zatim kad su ‘lokalni’ delovi niza sortirani, izlaskom iz pragme, master nit poziva jos jednom MergeSort nad celim nizom(novosortiranim od strane tc niti). Nakon toga se vrsi upis kompletno sortiranog niza u fajl res.txt. Zakomentarisana for petlja sluzi da se pokaze da je niz korektno sortiran, a zajedno sa svim ispisima je zakomentarisana radi preciznijeg merenja vremena. Na kraju se zaustavlja vreme I u milisekundama se prikazuje **vreme izvrsavanja koda**. Ubrzanje je prikazano u tabeli ispod. Na fakultetu sledece Vezbe cu pokrenuti za vise niti jer moj laptop nema vise od 4.

|  |  |  |
| --- | --- | --- |
| Number of threads – 10M  numbers | Execution time | Speed-Up |
| Sequential(Single) thread | 43330ms | - |
| Parallel code – 2 threads | 38239.8ms | 1.13 |
| Parallel code – 3 threads | 36623.9ms | 1.18 |
| Parallel code – 4 threads | 36289ms | 1.19 |

