

- **Rozbor:** tento radiaci algoritmus potrebuje lineárne pole  $p$  procesorov, v ktorom každý sa stará o  $n/p$  čísel. Pričom celkový počet čísel  $n$  musí byť väčší ako počet procesorov. Po  $p/2$  iteráciách je postupnosť zoradená. Jednotlivé iterácie prechádza predspracovanie, kedy každý procesor zoradí svoje čísla sekvenčným algoritmom. Každá iterácia sa skladá z dvoch častí, v prvej časti sa prepájajú procesory s nepárnym číslom, s procesormi s párnym číslom. V druhej časti je to naopak.

**Analýza:** predspracovanie  $O((n/p)\log(n/p))$ , prenos čísel  $O(n/p)$ , spojenie častí  $2 \cdot n/p$ , krok 1 alebo  $2 \cdot O(n/p)$ .

Časová náročnosť:  $O[(n/p) \log(n/p)] + O(n) = O((n \log n)/p) + O(n)$

Celková cena:  $c(n) = t(n) \cdot p = O(n \cdot \log n) + O(n \cdot p)$

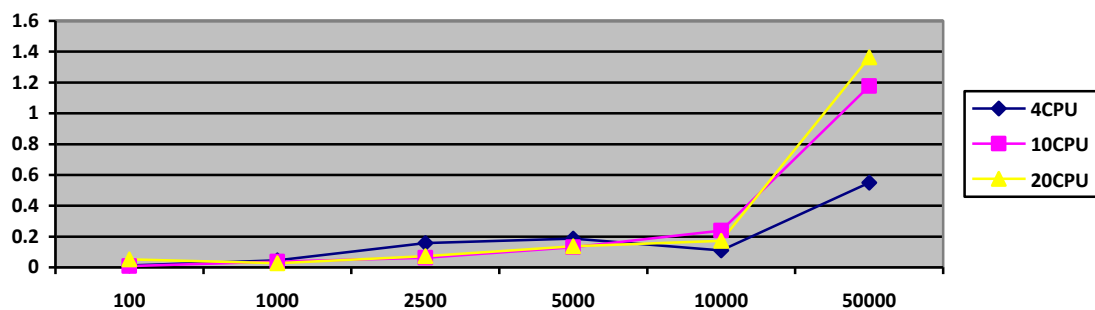
Priestorová náročnosť:  $P(n) = n$

Optimálnosť pre:  $p \leq \log n$

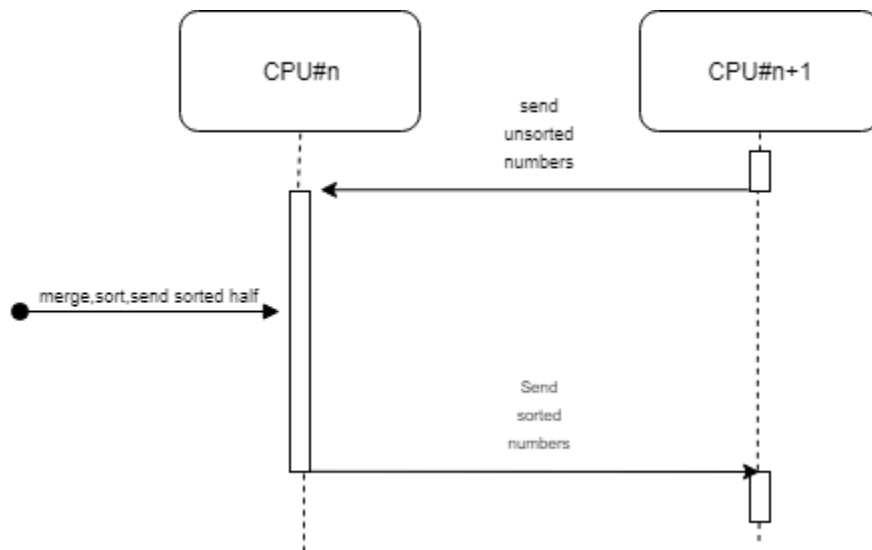
- **Implementácia:** Testovací skript test.sh vytvorí postupnosť náhodných čísel podľa zadaného parametru, ďalej preloží zdrojový súbor a spustí ho s daným počtom procesorov. Po spustení programu sa inicializuje prostredie, každému procesoru sa priradí jeho RANK(id) a určí sa celkový počet procesorov. Prvý procesor načíta súbor s náhodnými číslami a rozpošle príslušné čísla jednotlivým procesorom vrátane seba, pričom rozdelenie je čo najviac symetrické. Ak počet čísel nie je deliteľný počtom procesorov, zvyšok z tohoto delenia je rozdelený postupne od začiatku a od konca striedavo celému polu procesorov. Pri rozdeľovaní si pomáha vytvorenou tabuľkou distributionTable ktorá obsahuje index a počet priradených čísel procesoru. Rozdeľovanie prebieha v 3 fázach, najprv sa pošle počet čísel ktoré budú prenášané, nasleduje počet čísel ktoré má jeho sused napravo a za nimi už konkrétne priradené čísla. Každý procesor potom čaká na svoje dáta, a pokračuje sa na predspracovanie. Pri predspracovaní si každý procesor zoradí svoje čísla sekvenčným algoritmom. Po zistení počtu potrebných cyklov ( $p/2$ , ak  $p$  nie je deliteľné tak  $p/2+1$ ) nasleduje samotné paralelné radenie. V prvej časti cyklu začínajú radiť nepárne procesory, ktoré čakajú na čísla od pravého suseda. Následne zoradia spojenú postupnosť svojich čísel a čísel od suseda, potom pošlú zoradenú polovicu čísel susedovi napravo. V druhej časti cyklu zase párne procesory spolupracujú so svojim susedom po pravici. Po ukončení všetkých

cyklov čaká prvý procesor na všetky čísla od ostatných procesorov. Vypíše pôvodnú postupnosť čísel a následne zoradené postupnosti, ktoré dostal.

- Experimenty:** Pre meranie času bola použitá funkcia *MPI\_wtime()*. Meraný bol čas od začiatku načítania súboru po posledný výpis zoradenej postupnosti. Boli merané časy pre 100,1000,2500...50000 náhodných čísel, pričom ich radilo 4,10, a 20 procesorov. Najzaujímavejší sú posledné stĺpce pre 50000 kde je vidieť že optimálnosť algoritmu funguje iba v prípade že  $p \leq \log n$ , kde  $\log 50000 \approx 4.69$ , takže optimálny počet procesorov je **4 a menej**. Keď bolo použitých 10 a 20 procesorov tak sa algoritmus spomalil.



#### ▪ Komunikačný protokol:



- Záver:** Vzťah pre optimálnosť algoritmu sa potvrdil. Pri väčšom počte procesorov zaberá režia príliš veľku časť času potrebného pre dokončenie algoritmu. Vzhľadom na to že knižnica nie je určená pre praktické používanie môže táto skutočnosť veľmi ovplyvniť výsledky.