

Multiprocesorski sistemi

Domaći zadatak 3

MPI – osnove i komunikacija

Uvod

Cilj zadatka je da studente obučiti da samostalno podese MPI okruženje i razvijaju osnovne MPI programe.

Podešavanje okruženja

Preuzeti http://mups.etf.rs/vezbe/mpi/code/mpi_win_vs2008.zip (ili [..._vs2005.zip](http://mups.etf.rs/vezbe/mpi/code/mpi_win_vs2005.zip)). Prema uputstvima u priloženom `readme.txt` fajlu podesiti okruženje za razvoj i kontrolisano izvršavanje (engl. debugging) MPI programa na lokalnom računaru. Alternativno, koristiti OpenMPI na računaru `rtidev4.etf.rs`.

Zadaci

Svaki od programa treba napisati tako da može biti izvršen sa bilo kojim od broja procesa iz opsega navedenog iza postavke zadatka. N označava maksimalan mogući broj procesa u trenutno dostupnom MPI okruženju. Za programe koji će biti izvršavani na samo jednom računaru, pretpostaviti da važi $N=6$. Svaki program treba da vrši proveru da li je broj procesa tekućeg izvršavanja odgovarajući postavci zadatka. U slučaju da to nije zadovoljeno, prekinuti izvršavanje korišćenjem MPI poziva `Abort`.

Svaki program treba da ima ispis formatiran na sledeći način: rang procesa unutar `COMM_WORLD` komunikatora (2 cifre, sa vodećim nulama), znak `:`, redni broj poruke koju posmatrani proces ispisuje (3 cifre, sa vodećim nulama), znak `:`, tekst konkretne poruke. Na primer, u prvom zadatku proces sa rangom 0 će ispisati `00:001:Hello world!`, proces sa rangom 1 će ispisati `01:001:Hello world!` itd.

1. Sastaviti program koji ispisuje `Hello world!`. Od MPI poziva, program treba da sadrži samo `Init` i `Finalize`. [1,N]
2. Proširiti prethodni program tako da ispiše rang svog procesa i ukupan broj procesa u MPI svetu. [1,N]
3. Sastaviti program koji učitava broj u procesu sa rangom 0, pa taj broj šalje procesu sa rangom 1, koji ispisuje njegovu neizmenjenu vrednost. Koristiti MPI pozive `Send` i `Receive`. [2,2]
4. Sastaviti program koji učitava broj u procesu sa rangom 0, pa taj broj šalje ostalim procesima, koji ispisuju njegovu neizmenjenu vrednost. Koristiti MPI pozive `Send` i `Receive`. [2,N]
5. Sastaviti program koji učitava broj u procesu sa rangom 0, a zatim taj broj šalje svim procesima. Svaki od procesa treba da ispiše neizmenjenu vrednost poslatog broja. Koristiti MPI poziv `Bcast`. [1,N]
6. Sastaviti program koji određuje vrednost broja π . Proces sa rangom 0 treba da obavesti ostale procese o broju tačaka koje treba da obrade, prikupi podatke od ostalih procesa i ne treba da učestvuje u računanju. Za komunikaciju koristiti samo MPI pozive `Send` i `Receive`. [2,N]
7. Sastaviti program koji određuje vrednost broja π . Proces sa rangom 0 treba da obavesti ostale procese o broju tačaka koje treba da obrade, prikupi podatke od ostalih procesa i treba da učestvuje u računanju. Za komunikaciju koristiti samo MPI pozive `Bcast` i `Reduce`. [1,N]
8. Sastaviti program koji kvadrira sve elemente niza celih brojeva. Proces sa rangom 0 treba da obavlja svu komunikaciju sa korisnikom. Za komunikaciju koristiti MPI pozive `Scatter` i `Gather`. Ako broj elemenata niza nije celobrojni umnožak broja procesa u MPI svetu, prekinuti izvršavanje. [1,N]
9. Sastaviti program koji računa treći koren svakog elementa niza realnih brojeva. Proces sa rangom 0 (upravljač) treba da obavlja svu komunikaciju sa korisnikom i raspodelu poslova ostalim procesima (radnicima). Upravljač treba da šalje neobrađene elemente niza slobodnim radnicima i da prihvata rezultate od radnika, sve dok ima neobrađenih elemenata. Radnici treba da prihvataju po jedan element niza, obrađuju i vraćaju nazad sve dok ih upravljač ne obavesti da više nema posla. [2,N]

Važno: Ukoliko u nekom zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku i da nastavi da izgrađuje svoje rešenje na temeljima uvedene pretpostavke.

Napomena: Vrednost broja π se može statistički odrediti na više načina uz pomoć generatora pseudoslučajnih brojeva uniformne raspodele. Jedan od načina je generisanje tačaka u ravni sa obema koordinatama u realnom opsegu $[0,1]$. Vrednost broja π tada može biti određena na osnovu odnosa broja tačaka koje se nalaze u delu kruga poluprečnika 1 sa centrom u koordinatnom početku i broja tačaka koje pripadaju kvadratu stranice 1 koji obuhvata sve generisane tačke.