Prezime i ime: _____

Paralelno programiranje u raspodijeljenim sustavima

bodova / rješavao

[ ][ ] 1. (3) Zadan je MPI program (na slici desno). Svi procesi imaju lokalne varijable *a*, *b* i *c*, a *ID* je indeks pojedinog procesa. Koje vrijednosti će imati varijabla *c* za svaki proces na kraju izvođenja? Navedite sve mogućnosti (ako postoje). Obvezatno navedite (nacrtajte!) redoslijed izvođenja MPI operacija za svaki proces.

```
// Proces 1, ID = 1
MPI_Send(&ID, _, _, 2, _, _);
MPI_Recv(&a, _, _, MPI_ANY_SOURCE, _, _);
MPI_Send(&a, _, _, 3, _, _);
MPI_Recv(&b, _, _, MPI_ANY_SOURCE, _, _);
c = 2*a + b;
// Proces 2, ID = 2
MPI_Recv(&a, _, _, MPI_ANY_SOURCE, _, _);
MPI_Recv(&b, _, _, MPI_ANY_SOURCE, _, _);
c = 2*a + b;
MPI_Send(&c, _, _, 1, _, _);
MPI_Send(&ID, _, _, 3, _, _);
// Proces 3, ID = 3
MPI_Send(&ID, _, _, 2, _, _);
MPI_Recv(&a, _, _, MPI_ANY_SOURCE, _, _);
MPI_Recv(&b, _, _, MPI_ANY_SOURCE, _, _);
MPI_Send(&a, _, _, 1, _, _);
c = 2*a + b;
```

[ ][ ] 2. (3) Provesti *+_prescan* algoritam na zadanom polju duljine *n*=20 elemenata i na *p*=6 procesora. Označiti podjelu elemenata po procesorima, napisati izvedbu algoritma i provesti završni dio algoritma po procesorima. Ulazno polje je A[] = [4 7 1 0 5 2 6 4 8 1 9 7 3 0 1 5 2 7 4 3].

[ ][ ] 3. (3) Napisati algoritam za CRCW PRAM računalo koji će za zadano polje P[] stvoriti polje SP[] koje predstavlja rezultat provedbe jednostavnog sažimanja polja P[] gdje se za svaki podniz jednakih elemenata ulaznog polja stvara par oblika (vrijednost, broj_ponavljanja) (npr. za ulazno polje [3, 2, 2, 2, 1, 1] izlaz je [3, 1, 2, 3, 1, 2]). Za polje od n elemenata na raspolaganju je n procesora. Ocijeniti složenost algoritma.

[ ][ ] 4. (3) Paralelni algoritam iterativno računa elemente matrice. Nova vrijednost elementa računa se pomoću vrijednosti 2 neposredna susjedna elementa u svakom od 4 smjera, s tim da matrica ima 'spojene' sve bridove (npr. vrijednost elementa A[1,1] računa se pomoću A[1,2], A[1,3], A[1,N-1], A[1,N], A[N-1,1], A[N,1], A[2,1] i A[3,1]). Pretpostavite da stanje matrice omogućuje istovremeno računanje novih vrijednosti u jednoj iteraciji za sve elemente. Trošak računanja jednog elementa iznosi t_c. Izrazite trajanje izvođenja jedne iteracije na P procesora te učinkovitost i izoučinkovitost algoritma ako je matrica na procesore podijeljena:

a) po stupcima (svaki procesor ima jednak broj stupaca),

b) po podmatricama jednakih dimenzija.

Skicirajte komunikaciju među procesorima te obavezno navedite broj poruka po zadatku u jednoj iteraciji!

[ ][ ] 5. (2) Paralelno računalo plaća se 1 euro po satu po procesoru. Na raspolaganju nam je paralelni program čije se trajanje izvođenja može izraziti kao $T_p = 50 + 150/P$ (u satima). Dobit od rezultata izvođenja programa ovisi o trenutku dobivanja rezultata i opisana je izrazom $D = 18 \cdot (T_1 - T_p)$ (u eurima). Koje trajanje izvođenja nam donosi najveću moguću zaradu (dobit - troškovi) i na koliko procesora?

[ ][ ] 6. *Butterfly* komunikacijska struktura koristi se u slučaju kada svi procesi moraju imati rezultat na kraju provedbe paralelne operacije.

a. (1) Nacrtajte butterfly strukturu za slučaj 8 procesa (svakom procesu pridružen je jedan podatkovni element).

b. (2) Napišite odsječak programa (programsku petlju) kojim se ostvaruje ovaj način komunikacije među proizvoljnim brojem MPI procesa uporabom `Send` i `Recv` funkcija (komunikacija je u obliku hiperkocke). Program treba pronaći najveći podatak među svim procesima, a na kraju svi procesi moraju imati rezultat. Pretpostavke: broj procesa N je potencija broja 2, `ID` je indeks procesa a varijabla `x` je lokalni podatak svakog procesa.

[ ][ ] 7. (2) Korištenjem MPI funkcija Send i Recv napisati odsječak programa koji će za N procesa imitirati funkciju MPI_Barrier, tj. postići da svi procesi moraju doći do istog mjesta prije nego bilo koji proces može nastaviti s izvođenjem.

[ ][ ] 8. (1) Objasnite i skicirajte odnos pojmova 'procesor' - 'MPI proces' - 'zadatak u paralelnom algoritmu.