

## Četvrta laboratorijska vežba – OpenMP – 2022/23

1. Napisati sekvencijalni program kojim se vrši množenje dve kvadratne matrice A i B. Korišćenjem OpenMP direktive izvršiti paralelizaciju, tako da se izvrši distribucija iteracija petlje/petlji između niti. Izmeriti vreme množenja matrica za sekvencijalni slučaj i za slučaj posle uvođenja openMP direktiva. Uporediti oba rešenja za različite dimenzije matrica. Testirati za različit broj niti i različitu podelu iteracija između niti.

2. Napisati OpenMP kod koji sadrži sledeću petlju:

```
for (i=0;i<m;i++)  
    for (j=2;j<n;j++)  
        a[i][j]=2*a[i][j-2];
```

i izvršiti njenu paralelizaciju. Proučiti da li postoje zavisnosti između iteracija i po kom indeksu je moguća paralelizacija. Ispitati da li zamena petlji uzrokuje promenu u performansama.

3. Napisati OpenMP program kojim se pronalazi ukupan broj prostih brojeva između 1 i N, podelom iteracija petlje između različitih niti sa i bez korišćenja odredbe redukcije za kombinovanje parcijalnih rezultata u nitima. Uporediti vremena izvršenja u oba slučaja sa sekvencijalnim vremenom izvršenja. Uporediti ova rešenja za različite vrednosti N. Testirati za različit broj niti i različitu podelu iteracija između niti.

4. Napisati sekvencijalni program kojim se generiše skalarni proizvod dva vektora. Korišćenjem OpenMP direktive izvršiti paralelizaciju, podelom iteracija petlje nitima, sa i bez korišćenja odredbe redukcije za kombinovanje parcijalnih rezultata u nitima. Uporediti vremena izvršenja u oba slučaja sa sekvencijalnim vremenom izvršenja. Uporediti ova rešenja za različite dimenzije vektora. Testirati za različit broj niti i različitu podelu iteracija između niti.

5. Napisati OpenMP kod koji sadrži sledeću petlju:

```
for (i=1;i<m;i++)  
    for (j=0;j<n;j++)  
        a[i][j]=2*a[i-1][j];
```

i izvršiti njenu paralelizaciju. Proučiti da li postoje zavisnosti između iteracija i po kom indeksu je moguća paralelizacija. Ispitati da li zamena petlji uzrokuje promenu u performansama.

6. Napisati OpenMP program kojim se generiše prosečna vrednost elemenata vektora, podelom iteracija petlje nitima, sa i bez korišćenja odredbe redukcije za kombinovanje parcijalnih rezultata u nitima. Uporediti vremena izvršenja u oba slučaja sa sekvencijalnim vremenom izvršenja. Uporediti ova rešenja za različite vrednosti dimenzija vektora. Testirati za različit broj niti i različitu podelu iteracija između niti.

7. Napisati OpenMP program kojim se generiše maksimalna vrednost elemenata vektora, podelom iteracija petlje između različitih niti korišćenjem direktive *critical* za kombinovanje parcijalnih rezultata u nitima. Uporediti vremena izvršenja u oba slučaja sa sekvencijalnim vremenom izvršenja. Uporediti ova rešenja za različite vrednosti dimenzija vektora. Testirati za različit broj niti i različitu podelu iteracija između niti.