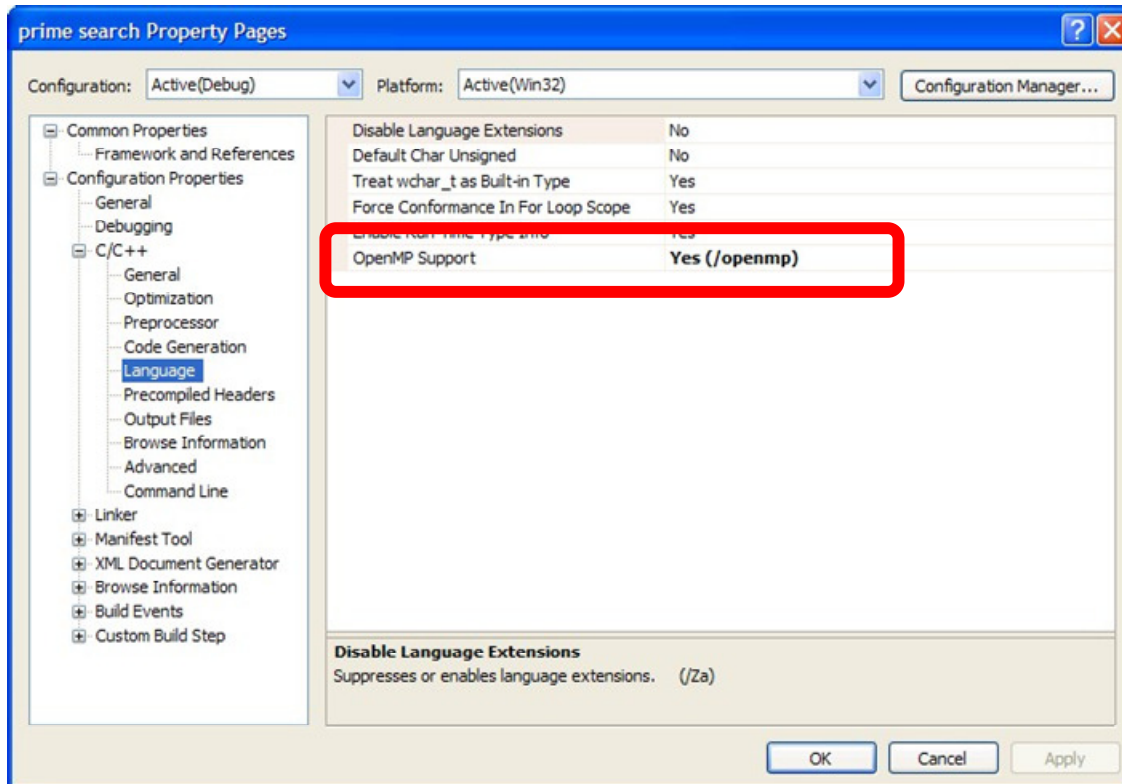


Podešavanje projekta:

1. Otvoriti **Project->properties** dijalog box
2. Selektovati **OpenMPsupport** iz **C/C++ -> Language**



Zadaci:

1. Napisati sekvencijalni program kojim se vrši množenje dve matrice A i B reda N. Korišćenjem OpenMP direktive izvršiti paralelizaciju petlje, tako da se izvrši distribucija iteracija između niti.

Podesiti broj niti tako da bude jednak broju jezgara računara. Izmeriti vreme množenja matrica za sekvencijalni slučaj i za slučaj posle uvođenja openMP direktiva. Uporediti oba rešenja za različite dimenzije matrica.

2. Napisati sekvencijalni program kojim se generiše skalarni proizvod dva vektora. Napisati OpenMP program kojim se generiše skalarni proizvod dva vektora, 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 dimenzije vektora.

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.

4. Napisati OpenMP program kojim se generiše prosečna vrednost elemenata vektora, podelom iteracija petlje između različitih niti sa i bez korišćenja naredbe 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.

5. 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.

6. Napisati OpenMP kod koji sadrži sledeću petlju:

```
for (i=0;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.