

## Exemple 2.1 : Boucle simple, versions séquentielle et parallèle

On calcule une boucle simple en versions séquentielle et parallèles avec OpenMP (4 versions)

### Examiner le code source

Dans un éditeur texte, ouvrir les fichiers :

- calcul\_seq.cxx
- calcul\_par0.cxx
- calcul\_par1.cxx
- calcul\_par2.cxx
- calcul\_par3.cxx

### Pour compiler:

Se mettre dans le répertoire qui contient de ce fichier Taper:

```
python build.py
```

Si tout s'est bien passé : deux fichiers exécutables ex\_seq\_2\_1 et ex\_par\_2\_1 sont créés dans le répertoire build.

ex\_par\_2\_1 exécute successivement les 4 versions parallèles

---

### Pour exécuter (6 threads pour la version parallèle):

Taper :

```
./build/ex_seq_2_1 n  
  
OMP_NUM_THREADS=6 ./build/ex_par_2_1 n
```

où n est un entier positif (taille des vecteurs)

si n n'est pas spécifié, le code prend n = 1000

Comparer les resultats et les temps d'exécution des différentes versions