**Travail à réaliser - unité “parallélisme” :**

1. L'objectif est de se mettre dans le contexte d’une application simple pour par la suite appliquer la méthodologie d’optimisation des calculs / parallélisation à l’aide de OpenMP.
2. Application à réaliser : comptage de personnes dans un contexte simple.
   1. Les étapes à réaliser (en C/C++, OpenCV, OpenMP) sont :
      1. Lecture de la vidéo
      2. Calcul de l’arrière plan qui s’actualise à chaque image (algorithme d’estimation et de soustraction de l’arrière plan)
      3. Calcul de la différence entre l’image d’arrière plan et l’image acquise, le résultat sera seuillé pour détecter le mouvement significatif (à régler à la main, pour que ca marche bien)
      4. Filtrage par ouverture morphologique si nécessaire
      5. Comptage des composantes connexes restantes (de manière la plus simple possible)
3. Réaliser le profiling de l’application (gettimeofday recommandé), gprof ne marche pas bien avec OpenCV
4. Pour la fonction/étape du traitement qui est la plus coûteuse en temps de calcul, essayez d’optimiser - par changement d’algorithme ou par l’utilisation des directives OpenMP. Si ce n’est pas possible, argumentez et choissiez une fonction à optimiser plus facile.
5. Évaluez les gain en fonction des types d’optimisation et du nombre de processeurs utilisés si applicable.
6. Rédiger un rapport. Le rapport doit contenir :
   1. Page de garde, Introduction, présentation de l'application et discussion des aspects de l’implémentation
   2. Présentation des ressources de calculs (discuté en séance)
   3. Profiling d’une première implémentation
   4. Choix de la fonction à optimiser, description de la stratégie de parallélisation, représentée graphiquement comment présenté en cours
   5. Evaluation et validation des résultats calcul des speedup et des speedups relatifs, graphiques, conclusions
   6. Conclusion générale
7. Vous pouvez travailler en binôme.
8. Évaluation sera faite sur la base du rapport et les codes joints en annexe.
9. La date de rendu : 15/04/2023