

INITIATION AU CALCUL DISTRIBUÉ AVEC PYTHON

— Public	Développeurs Python, administrateurs systèmes souhaitant automatiser des tâches, ingénieurs et data scientists en quête de puissance de calcul, développeurs de sites Internet à forte charge
— Durée	5 jours - 35 heures
— Pré-requis	Pratique régulière du langage Python et si possible des bibliothèques numpy et pandas.
— Objectifs	Comprendre les concepts de la programmation parallèle Savoir identifier quelles parties d'un programme sont parallélisables et identifier les facteurs limitant les performances de votre programme Posséder une vue d'ensemble de l'écosystème de calcul parallèle de Python Posséder les bases de la programmation parallèle: multithreading et multiprocessing Savoir distribuer des calculs sur plusieurs ordinateurs Savoir calculer sur GPU Savoir utiliser les frameworks modernes dont Dask, Vaex, Xarray, Ray ou Daskxarray pour le calcul numérique Savoir créer des workflows de traitements de données avec Prefect
— Méthodes pédagogiques	Pour bien préparer la formation, le stagiaire remplit une évaluation de positionnement et fixe ses objectifs à travers un questionnaire. La formation est délivrée en présentiel ou distanciel (e-learning, classe virtuelle, présentiel et à distance). Le formateur alterne entre méthodes démonstratives, interrogatives et actives (via des travaux pratiques et/ou des mises en situation). La validation des acquis peut se faire via des études de cas, des quiz et/ou une certification. Cette formation est animée par un consultant-formateur dont les compétences techniques, professionnelles et pédagogiques ont été validées par des diplômes et/ou testées et approuvées par l'éditeur et/ou par Audit Conseil Formation.
— Moyens techniques	1 poste de travail complet par personne De nombreux exercices d'application Mise en place d'ateliers pratiques Remise d'un support de cours Passage de certification(s) dans le cadre du CPF Remise d'une attestation de stage
— Modalité d'évaluation des acquis	Evaluation des besoins et objectifs en pré et post formation Evaluation technique des connaissances en pré et post formation Evaluation générale du stage
— Délai d'accès	L'inscription à cette formation est possible jusqu'à 5 jours ouvrés avant le début de la session
— Accessibilité handicapés	Au centre d'affaires ELITE partenaire d'ACF à 20 m. Guide d'accessibilité à l'accueil.

JOUR 1 : LES CONCEPTS DE LA PROGRAMMATION PARALLÈLE

1. LES DIFFÉRENTES FORMES DE LA PROGRAMMATION PARALLÈLE

- Au sein d'un même programme: multithreading, SIMD, compilation Just In Time
- Sur une même machine: multiprocessing, calcul distribué
- Sur plusieurs machines: calcul distribué, HPC
- Bien comprendre les différences entre multithreading, multiprocessing et programmation asynchrone

2. LES BASES DE LA PROGRAMMATION PARALLÈLE

- Multithreading: mise en oeuvre, gestion des accès concurrents
- Multiprocessing: mise en oeuvre, mémoire partagée, pools de process
- Présentation des différentes primitives associées au multiprocessing et multithreading: sémaphores, files d'attente, conditions, barrières, ...
- Comprendre les limites du multithreading en Python: Le GIL et savoir le désactiver

3. PRÉSENTATION DE L'ÉCOSYSTÈME DE CALCUL PARALLÈLE DE PYTHON

- Panorama des différentes librairies par thématique: Just In Time Compilation, multithreading, Multiprocessing, Calcul distribué, Grille de calcul, Calcul sur GPU...

JOUR 2 MATIN : PROFILING ET OPTIMISATION

- Monitorer les performances de votre programme : les différents types de profileurs et leur mise en oeuvre
- Créer son propre profileur
- Identifier ses goulots d'étranglement, savoir mesurer les performances de votre matériel :
- disque dur, mémoire, cpu
- Pensez à optimiser votre code avant de le paralléliser : techniques et algorithmes
- Compilation Just in Time avec Pypy et numba
- Logger et déboguer vos process avec Eliot

JOUR 2 APRÈS MIDI : CALCUL DISTRIBUÉ

- Calcul distribué avec Celery et dramatiq: exécuter des tâches en arrière plan dans vos programmes Django avec Dramatiq
- Ray: distribuez vos tâches avec simplicité et performance

JOUR 3 : BIG DATA

- Vaex: manipulez des données tabulaires en utilisant toute la puissance de votre ordinateur et les algorithmes les plus performants
- Dask: manipulez des données numériques ou tabulaires qui ne tiennent pas en RAM en les parallélisant automatiquement sur votre ordinateur portable, réseau local ou supercalculateur
- XArray: manipulez les fichiers scientifiques et big data ((NetCDF, HDF5, parquet...)) avec facilité et en bénéficiant de tous les avantages de Numpy, Pandas et Dask
- Visualiser des données volumineuses avec DataShader et Holoviz

JOUR 4 : CALCULER SUR GPU

- Comprendre l'architecture matérielle d'un GPU
- Savoir choisir son matériel pour calculer sur GPU avec Python
- Panorama des différentes librairies de Calcul sur GPU avec Python
- Comprendre quand un GPU est préférable à un CPU et quand il ne l'est pas
- Compilation Just In Time sur GPU avec Numba
- Calcul numérique avec Cupy
- Manipuler les dataframes Pandas avec RapidsAI

JOUR 5 : CRÉER DES WORKFLOWS DE TÂCHES AVEC PREFECT : 0,5 JOUR

- Présentation des concepts: tâches, paramètres, flows...
- Créer un workflow de traitement de données: téléchargement, filtres, transformations, sauvegarde
- Gérer les erreurs et reprises sur incidents
- Superviser l'exécution de votre workflow

NOUS CONTACTER

Siège social

16, ALLÉE FRANÇOIS VILLON
38130 ÉCHIROLLES

Téléphone

04 76 23 20 50 - 06 81 73 19 35

Suivez-nous sur les réseaux sociaux, rejoignez la communauté !



ACF Audit Conseil Formation



@ACF_Formation

Dernière mise à jour : 15/04/2026

PROFIL Formateur : Les formateurs sont recrutés selon plusieurs critères :
Expérience, pédagogie, dynamisme et prévoyance.