

Corso di **High-Performance e Parallel Computing per l'Astrofisica**

3 MODULI

tra Ottobre 2026 e Aprile 2027



INAF
ISTITUTO NAZIONALE
DI ASTROFISICA



Durata

- 120 ore complessive
- 3 moduli da 40 ore
- 5 giorni, 4+4 ore al giorno

Obiettivi formativi

Il corso fornisce una competenza completa e operativa nel calcolo ad alte prestazioni e nella programmazione parallela, dal funzionamento delle architetture di calcolo moderne fino allo sviluppo, all'ottimizzazione, al debugging e al profiling di codice parallelo su sistemi multi-core, multi-nodo e accelerati da GPU.

Profilo in uscita: una figura capace di portare un'applicazione scientifica o ingegneristica dal prototipo seriale a una versione parallela, efficiente, scalabile e riproducibile su sistemi HPC eterogenei.

A chi si rivolge

Il corso si rivolge principalmente a:

- **Studenti undergraduate e graduate** che vogliono acquisire competenze di calcolo parallelo non coperte dai curricula standard;
- **Dottorandi e post-doc** in fase iniziale di carriera nelle scienze computazionali (fisica, astrofisica, scienza dei materiali, ingegneria, life sciences, data science), che hanno bisogno di portare i propri codici su sistemi HPC;
- **Professionisti già affermati** (ricercatori, ingegneri, Research Software Engineer) che vogliono approfondire o aggiornare le proprie competenze in HPC e programmazione parallela.

È particolarmente utile a chi sviluppa o mantiene codice scientifico ad alta intensità di calcolo o di dati e ha la necessità concreta di farlo scalare in modo efficiente.

Struttura modulare e livello:

I tre moduli sono pensati per essere fruibili anche separatamente: un partecipante esperto può iscriversi al solo Modulo B o C (verificati i prerequisiti), mentre uno studente alle prime armi seguirà l'intero percorso in sequenza. Le hands-on session prevedono esercizi su due livelli (base e stretch) per gestire la varietà di esperienza in aula.

Requisiti

Prerequisiti generali (tutti i moduli)

- Familiarità con l'ambiente Linux e l'uso della riga di comando (shell, editor, compilazione, gestione dei file).
- Familiarità con il linguaggio C/C++ (compilazione, puntatori, gestione della memoria, strutture dati di base).
- È utile, ma non indispensabile, una conoscenza di base di analisi numerica e algebra lineare.

Prerequisiti per-modulo (per chi si iscrive ai singoli moduli)

- Modulo A: solo i prerequisiti generali.
- Modulo B: contenuti del Modulo A o competenze equivalenti (paradigma a memoria condivisa, nozioni di architettura del nodo).
- Modulo C: contenuti dei Moduli A e B o competenze equivalenti (MPI di base/intermedio e programmazione GPU di base).

Contenuti didattici

MODULO A – Architetture moderne, codice seriale e memoria condivisa (40h)

Obiettivi formativi: comprendere come le scelte di codice si traducano in performance sulla microarchitettura; saper portare un kernel seriale a una versione vettorizzata e cache-friendly; saper parallelizzare su nodo singolo con OpenMP gestendo correttamente la memoria condivisa; saper individuare bug di correttezza e colli di bottiglia seriali con strumenti standard.

Contenuti didattici

Architetture moderne (CPU)
Ottimizzazione del codice seriale
Architettura del nodo e NUMA
Multithreading con OpenMP
Debugging & Profiling I (seriale)

Competenze in uscita: il partecipante sa analizzare e ottimizzare codice seriale guidato da misure, e sa scrivere codice OpenMP corretto ed efficiente su architetture NUMA.

MODULO B – Memoria distribuita (MPI) e introduzione alle GPU (40h)

Obiettivi formativi: comprendere il modello a memoria distribuita e saper scrivere applicazioni MPI corrette ed efficienti fino al livello intermedio; padroneggiare le tecniche MPI avanzate per pattern di comunicazione complessi e I/O parallelo; saper fare debugging e profiling di applicazioni parallele multi-process.

Contenuti didattici

- Architetture dei cluster
- MPI – base
- MPI – intermedio
- MPI – avanzato
- Debugging & Profiling II (parallelo)
- Containerizzazione per HPC

Competenze in uscita: il partecipante sa parallelizzare un'applicazione su più nodi con MPI e sa progettare e ottimizzare applicazioni parallele complete su sistemi HPC eterogenei e su larga scala, scegliendo pattern algoritmici e di comunicazione adeguati, e sa renderne riproducibile l'esecuzione. Inoltre sa profilare e debuggare i propri codici su piattaforme parallele.

MODULO C – GPU, algoritmi e deployment (40h)

Obiettivi formativi: comprendere l'architettura GPU e saper scrivere e ottimizzare kernel di base sfruttando il memory model; saper gestire scenari multi-GPU e multi-nodo con communication libraries dedicate; saper profilare applicazioni alla scala di produzione e distribuirle in modo riproducibile.

Contenuti didattici

- GPU I
- GPU II (avanzata e distribuita)
- Profiling su larga scala

Competenze in uscita: il partecipante sa scrivere e ottimizzare kernel di base e portarli su GPU ottimizzandone gli accessi in memoria, misurandone le prestazioni con strumenti di profiling avanzato. Sa utilizzare cluster multi-nodo e multi-GPU programmando i propri codici in modo da utilizzare un approccio misto MPI+GPU

Materiale didattico

- Slide delle lezioni frontali per ciascun modulo.
- Notebook ed esercitazioni hands-on con tracce e soluzioni di riferimento.
- Repository Git con codici di esempio, scheletri di esercizio e soluzioni.
- Accesso a risorse di calcolo HPC (CPU, multi-nodo e GPU) per le sessioni pratiche, con istruzioni d'ambiente (moduli, scheduler, container).
- Dispense sintetiche e bibliografia/riferimenti per approfondimento.

Profilo docente

Il corso è tenuto da un team di docenti con esperienza diretta di ricerca e sviluppo in ambito HPC e calcolo scientifico, attivi nello sviluppo e nell'ottimizzazione di codici di simulazione su sistemi di classe pre-exascale ed exascale.

- **Luca Tornatore** – Astrofisico ed esperto di HPC presso INAF (Trieste); Expertise: architetture, MPI, profiling, algoritmi
- **Antonio Ragagnin** – Ricercatore presso INAF Trieste, Expertise: OpenMP/MPI/GPU/algoritmi
- **Giuliano Taffoni** – Development Scientist presso INAF Trieste, Expertise: OpenMP, GPU offloading, containerizzazione
- **David Goz** – Ricercatore presso INAF Trieste, Expertise: OpenMP/MPI/GPU, memory model, profiling
- **Giovanni Lacopo** – Ricercatore Tecnologo presso INAF Trieste, Expertise: MPI, GPU AMD, nuove architetture, comunicazione distribuita su GPU
- **Valentina Cesare** – Tecnologo presso INAF Catania, Expertise: GPU

Modalità di erogazione

Formazione in presenza, con forte componente pratica su risorse HPC reali. Ogni giornata alterna lezione frontale e hands-on (4+4 ore).

Calendario

Modulo	Date	Sede
Modulo A	19–23 ottobre 2026	Università degli Studi dell'Insubria Chiostro di Sant'Abbondio - Via Sant'Abbondio 12, Como
Modulo B	30 novembre – 4 dicembre 2026	Università degli Studi dell'Insubria Chiostro di Sant'Abbondio - Via Sant'Abbondio 12, Como
Modulo C	5 – 9 aprile 2027	CNR Area Territoriale di Ricerca di Bologna Via Piero Gobetti 101 - 40129 Bologna

Attestato

Al termine di ciascun modulo viene rilasciato un attestato di frequenza del singolo modulo.

Iscrizione

La fee di iscrizione è di **100€ a persona per ogni modulo**, i tre moduli sono pensati per essere fruibili anche separatamente. La quota di partecipazione non comprende le spese di viaggio e di alloggio, che restano interamente a carico dei partecipanti. Sono invece inclusi nella quota il pranzo e i coffee break previsti durante tutte le giornate di lezione.

Le iscrizioni sono aperte al seguente link:

https://agenda.supercomputing-icsc.it/e/HPC_e_parallel_computing_per_astrofisica

I posti disponibili sono 40 e l'ammissione in aula avverrà secondo l'ordine di registrazione. Qualora il numero di iscrizioni superi i posti disponibili, alla chiusura delle registrazioni avranno priorità dottorandi/e e giovani ricercatori/trici, sempre nel rispetto dell'ordine temporale di iscrizione