

Calendario de clases

IMT2112 - Algoritmos Paralelos en Computación Científica

Elwin van 't Wout

10 de agosto de 2020

A continuación se presenta el programa del curso IMT2112 del segundo semestre de 2020. Cabe señalar que el orden y contenido de las clases y evaluaciones puede cambiar durante el semestre.

Modalidad

Las clases, ayudantías e interrogaciones de este curso son virtuales. No habrá actividades presenciales. Para las clases y ayudantías usaremos Zoom. La plataforma de Canvas será usado para anuncios, materiales de clase y las interrogaciones. Es responsabilidad del estudiante estar pendiente de los anuncios en Canvas.

El horario de clases es martes y jueves, módulo 2. El horario de la ayudantía es viernes, módulo 2.

Las clases tendrán el formato siguiente:

- 10:00–10:10 h, consultas;
- 10:10–11:10 h, cátedra;
- 10:10–11:20 h, consultas.

Los estudiantes pueden participar activamente en la cátedra, haciendo consultas y observaciones.

Calendario

| Clase | Fecha | Contenido | Bibliografía |
|-------|---------|-------------------------------|-----------------------|
| C.1 | 11-8-20 | Introducción | Presentación |
| C.2 | 13-8-20 | Multiprocessing | Eijkhout 1.1 |
| A.1 | 14-8-20 | Python multiprocessing | |
| C.3 | 18-8-20 | Computer architectures | Eijkhout 1.1 |
| C.4 | 20-8-20 | Instruction-level parallelism | Eijkhout 1.2 |
| A.2 | 21-8-20 | Python multiprocessing | |
| C.5 | 25-8-20 | Memory hierarchies | Eijkhout 1.3 |
| C.6 | 27-8-20 | Data locality | Eijkhout 1.4–1.6 |
| A.3 | 28-8-20 | C++ | |
| C.7 | 1-9-20 | Parallel efficiency | Eijkhout 2.1–2.3 |
| C.8 | 3-9-20 | Granularity | Eijkhout 2.4+2.5+2.10 |
| A.4 | 30-8-20 | OpenMP | |
| C.9 | 8-9-20 | Threading | Eijkhout 2.6.1-2.6.2 |
| C.10 | 10-9-20 | Collective operations | Eijkhout 6.1 |
| A.5 | 11-9-20 | OpenMP | |
| I.1 | 15-9-20 | Interrogación 1 | Clases 1–8 |
| C.11 | 17-9-20 | Parallel matvec | Eijkhout 6.2 |
| | 18-9-20 | <i>Feriado</i> | |
| | 22-9-20 | <i>Suspensión</i> | |
| | 24-9-20 | <i>Suspensión</i> | |
| | 25-9-20 | <i>Suspensión</i> | |

| Clase | Fecha | Contenido | Bibliografía |
|-------|----------|--------------------------------|----------------|
| C.12 | 29-9-20 | Parallel LU | Eijkhout 6.3 |
| C.13 | 1-10-20 | Parallel matmat | Eijkhout 6.4 |
| A.6 | 2-10-20 | Clúster | |
| C.14 | 6-10-20 | Message Passing | Eijkhout 2.6.3 |
| C.15 | 8-10-20 | MPI statics | Eijkhout 2.6.3 |
| A.7 | 9-10-20 | MPI | |
| C.16 | 13-10-20 | MPI communication | Eijkhout 2.6.3 |
| C.17 | 15-10-20 | Diferencias finitas | Apuntes |
| A.8 | 16-10-20 | MPI | |
| C.18 | 20-10-20 | Parallel sparse linear algebra | Eijkhout 6.5 |
| I.2 | 22-10-20 | Interrogación 2 | Clases 9–16 |
| A.9 | 23-10-20 | MPI | |
| C.19 | 27-10-20 | Parallel linear solvers | Eijkhout 6.6 |
| C.20 | 29-10-20 | Parallel preconditioners | Eijkhout 6.7 |
| A.10 | 30-10-20 | OpenCL | |
| C.21 | 3-11-20 | Colouring strategies | Eijkhout 6.8 |
| C.22 | 5-11-20 | Heterogeneous computing | Presentación |
| A.11 | 6-11-20 | OpenCL | |
| C.23 | 10-11-20 | GPU computing | Presentación |
| C.24 | 12-11-20 | OpenCL | Presentación |
| A.12 | 13-11-20 | OpenCL | |
| C.25 | 14-11-20 | OpenCL | Presentación |
| C.26 | 16-11-20 | Ordenar areglos | Presentación |
| A.13 | 20-11-20 | OpenCL | |
| C.27 | 24-11-20 | HPC | Presentación |
| C.28 | 26-11-20 | HPC | Presentación |
| A.14 | 27-11-20 | OpenCL | |
| C.29 | 1-12-20 | Repaso | |
| I.3 | 3-12-20 | Interrogación 3 | Clases 17–28 |

Evaluación

El curso contempla las evaluaciones siguientes.

1. Quizzes, regularmente durante el semestre.
2. Interrogación 1, 15-9-2020, 10:00 h.
Clases 1–8.
3. Interrogación 2, 22-10-2020, 10:00 h.
Clases 9–16.
4. Interrogación 3, 3-12-2020, 10:00 h.
Clases 17–28.
5. Tarea 1, 2-9-2020.
Python multiprocessing.
6. Tarea 2, 30-9-2020.
OpenMP threading.
7. Tarea 3, 28-10-2020.
MPI distributed parallelisation.
8. Tarea 4, 25-11-2020.
OpenCL GPU computing.

Los lineamientos para las evaluaciones del curso son el siguiente.

- La nota final es el promedio de las ocho evaluaciones, con ponderación uniforme.
- Los quizzes y las interrogaciones son individuales. Los estudiantes no pueden interactuar entre ellos durante las interrogaciones.
- Las tareas de programación deben ser preparados y entregados de forma individual. El código debe ser escrito por el estudiante: no se puede copiar código de compañeros, tampoco usar código de fuentes externos tales como el internet. Los estudiantes sí pueden conversar entre ellos sobre las tareas de programación y compartir experiencias, siempre cuando el código entregado es escrito por su mismo.

Integridad académica

Todos los participantes en el curso (profesor, ayudante y estudiantes) deben cumplir con los estándares comunes de la integridad académica, en particular el Código de Honor UC.

Bibliografía

El libro principal del curso es:

1. Eijkhout, Victor. “*Introduction to High Performance Scientific Computing*,” third edition, TACC, 2020. Online version of 23-1-2020.

Más información se puede encontrar en los libros:

1. Eijkhout, Victor. “*Introduction to Scientific Programming in C++/Fortran2003*,” TACC, Feb. 2020.
2. Eijkhout, Victor. “*Parallel Programming in MPI and OpenMP*,” TACC, 2017.

Estos libros son de fuente abierto y libremente disponible en:

<http://www.tacc.utexas.edu/~eijkhout/istc/istc.html>