Parallel Computing
- Course Information -

Dr. Mario Garza Fabre
mario.garza@cinvestav.mx

Cinvestav Unidad Tamaulipas
Ciudad Victoria, Tamaulipas, México

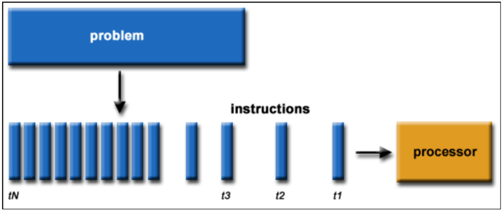
January - April, 2023

Contents

1. Introduction / Course Information
2. Assessment
3. Project Deliverables
4. Project Topic
5. Infrastructure

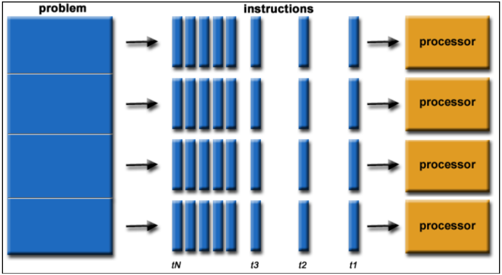
Why Parallel Computing? Why Should We Study It?

Why Parallel Computing? Why Should We Study It?



Serial Computing
Instructions are executed sequentially one after another by single processor.

Parallel Computing
Instructions execute simultaneously on different processors (an overall control/coordination mechanism is employed).



Why Parallel Computing? Why Should We Study It?

- Computational power is increasing, but so are our problems:
  - Complex problems are still waiting to be solved
  - We need answers/solutions in reasonable time
  - We need to process large volumes of data in an efficient manner
- Parallelism is a natural way of improving performance
- Adding more processors doesn't help much if programmers aren't aware of them (or don't know how to use them)!

What About This Course?

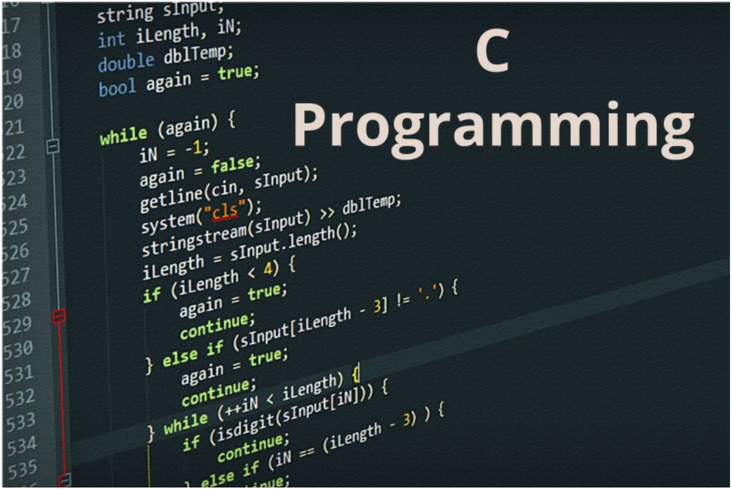
We will focus on **techniques and applications** of parallel computing, as well as on the ways to **write computer programs** using this paradigm.

Contents

- Introduction to Parallel Computing (Hardware and Software)
- Parallel Programs / Programming for Performance
- General Strategies of Parallel Computing
- Algorithms and Applications

- Shared-Memory Programming
  - Pthreads
  - OpenMP
- Distributed-Memory Programming
  - Message-Passing Interface (MPI)

Prerequisites



```
string sInput;
int iLength, iN;
double dblTemp;
bool again = true;

while (again) {
    iN = -1;
    again = false;
    getline(cin, sInput);
    system("cls");
    stringstream(sInput) >> dblTemp;
    iLength = sInput.length();
    if (iLength < 4) {
        again = true;
        continue;
    } else if (sInput[iLength - 3] != '.') {
        again = true;
        continue;
    } while (++iN < iLength) {
        if (isdigit(sInput[iN])) {
            continue;
        } else if (iN == (iLength - 3) ) {
```

Bibliography

Main textbook:

- **An Introduction to Parallel Programming.** Peter Pacheco, Morgan Kaufmann (Elsevier), 2011

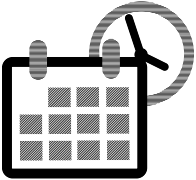
Additional books:

- **An Introduction to Parallel Computing: Design and Analysis of Algorithms.** A. Grama *et al.*, Addisson-Wesley, 2nd Ed, 2003
- **Parallel Programming in C with MPI and OpenMP.** Michael J. Quinn, McGraw-Hill, 2003

Online resources:

- **Lawrence Livermore National Laboratory** online tutorials. `https://hpc.llnl.gov/training/tutorials`

Course Information: Dates / Times



4 January - 25 April, 2023

	Time	**Room**
Tuesday:	10:00 - 12:00	A-255
Friday:	10:00 - 12:00	A-255

Course Information: Materials and Submissions

Course materials/resources will be available at:
https://www.dropbox.com/sh/w9feorc6hypfivc/AAD9n6DO2_NdhvBJqPsRL0m4a?dl=0

Submitting assignments/deliverables:

- A **Dropbox** folder will be shared with each student
- For each assignment/deliverable, create a **sub-folder** and place files inside
- **Do not send things by email!!!**

Assessment / Grading

Projects (5 deliverables)	80%
Presentation(s)	10%
Final exam	20%
Total	**110 %**

NOTE: if no "Final exam", then the value of "Projects" will be 100%.

Projects / Deliverables

- Each student will **INDIVIDUALLY execute a project**
- **A different project will be assigned to each student**
- Potential project topics will be discussed at the end of this session
- Project **involves the following deliverables:** (tentative dates)

Deliverable	Deadline	Value (%)
Sequential version	**27 January**	20%
Shared-memory version (Pthreads)	**24 February**	20%
Shared-memory version (OpenMP)	**17 March**	20%
Distributed-memory version	**7 April**	20%
Hybrid (shared/distributed) version	**21 April**	20%

Important Remarks

- **Deadlines**
  - Assignments: to be considered **iff** received by the deadline
  - Projects: value reduced **10% every 24 hours** of delay
- **Completeness**
  - Receive 100% of value if meet **100% of requirements**
  - Otherwise, value will be **proportional**
- **Individual Work and Teamwork**
  - **You will work individually** (unless otherwise indicated)
  - When working as a team, **assessment is still individual**

Assessment Criteria for Presentations

Assessment of presentations will consider:

- Contents and organization
- Knowledge / understanding
- Ability / effort to transmit knowledge
- Formality (behavioral)
- Quality of materials
- Time management

Attendance Policies

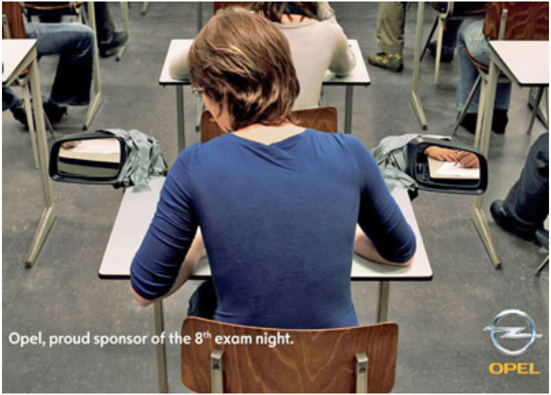
- Attendance is not a criterion for grading
- **Mandatory:** 100% attendance required to pass
- **Be on time:** 15 minute tolerance
- Absences need to be formally justified (through **Academic Tutor** or **Academic Coordinator**)

Ethics and Behaviour

Ethics and Behaviour - Honesty

Honesty will be crucial in this course!!!

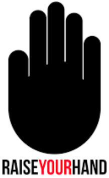
WHAT IS PLAGIARISM?



Cheating will cause **COURSE FAILURE!**

Ethics and Behaviour - Respect

- Instructor must respect students
- Students must respect instructor
- Students must respect classmates



- Avoid interruptions!
- Raise your hand!
- No phones!
- Be on time!

Exploit This Opportunity

- A graduate course is one of the **last chances to strengthen and expand our formation**
- **Contents are ambitious**, topics can't be covered with desired depth
- A graduate student must be able to **work and learn independently**
- **Do not limit yourself** to the materials seen in class
- A graduate course **assumes certain knowledge** / skills
- **Identify your weaknesses and work hard to overcome them!**

Project Deliverable 1: Sequential Version

You will submit the **source code** of your implementation **(70%)** and a **LaTeX report** with the following contents **(30%)**:

1. Title, author name
2. Abstract
3. Introduction
    - Problem / Task / Topic Description
4. Serial Algorithm
    - Pseudocode, explanation of functioning, complexity analysis
5. Results
    - Examples / verification of results
    - Plot + table: execution time vs problem size (5+ different sizes)
    - Report the fastest time observed in 10 independent executions
6. Conclusions
7. References

Results MUST be **interpreted** / **discussed!** (not enough to include plots)

Selection of a Project Topic: Rules

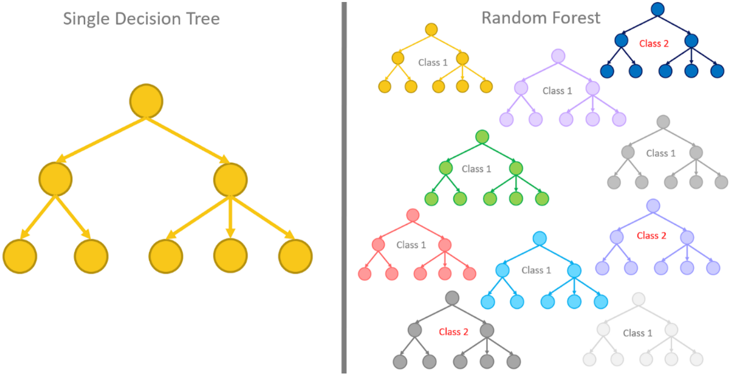
- A **different project** will be **INDIVIDUALLY** executed by each student
- Some project options have been pre-assigned to each of you
  - Make your **final choice** by **Friday, January 13, 2023**
  - Communicate this by placing a text file (**PROJECT.txt**) in your Dropbox folder with your decision(s)
  - **No changes after this** (unless discussed and authorized)
- **You will be the only responsible** for investigating and understanding the problem/algorithm, as well as for the design and implementation of the serial/parallel programs

Abraham - Parallelization of Optimization Algorithms

- Option 1: **Branch and Bound**
  - `https://en.wikipedia.org/wiki/Branch_and_bound`
- Option 2: **Population-based Metaheuristics**
  - `https://en.wikipedia.org/wiki/Metaheuristic`
- **Use the chosen algorithm to solve a problem such as:**
  - Graph coloring, quadratic assignment problem, ...
  - Any NP-complete problem (`https://en.wikipedia.org/wiki/List_of_NP-complete_problems`)
  - A particular problem you are interested in!

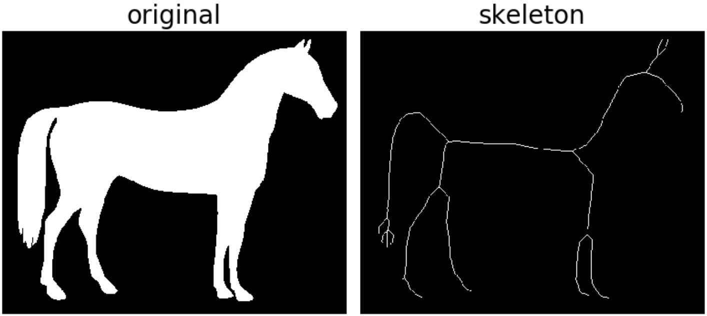
Michel - Option 1: Ensemble Machine Learning Methods

Random Forests Classification Algorithm



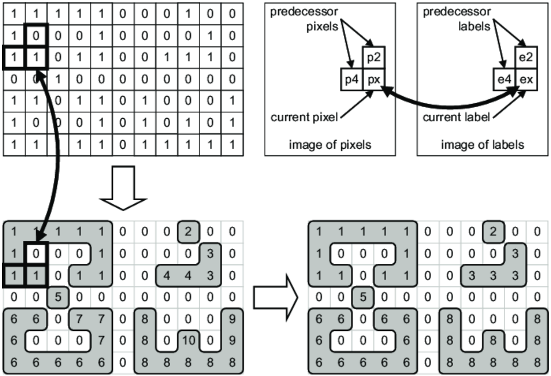
```
https://en.wikipedia.org/wiki/Random_forest
https://en.wikipedia.org/wiki/Ensemble_learning
https://builtin.com/data-science/random-forest-algorithm
https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm
```

Michel - Option 2: Image Thinning (Skeletonization)



https://homepages.inf.ed.ac.uk/rbf/HIPR2/thin.htm
https://scikit-image.org/docs/dev/auto_examples/edges/plot_skeleton.html
https://boofcv.org/index.php?title=Example_Morphological_Thinning
https://reference.wolfram.com/language/ref/Thinning.html?view=all
https://rosettacode.org/wiki/Zhang-Suen_thinning_algorithm

Michel - Op. 3: Connected-Component Labeling (Region Labeling)



1	1	1	1	1	0	0	0	1	0	0
1	0	0	0	1	0	0	0	0	1	0
1	1	0	1	1	0	0	1	1	1	0
0	0	1	0	0	0	0	0	0	0	0
1	1	0	1	1	0	1	0	0	0	1
1	0	0	0	1	0	1	0	1	0	1
1	1		1	1	0	1	1	1	1	1

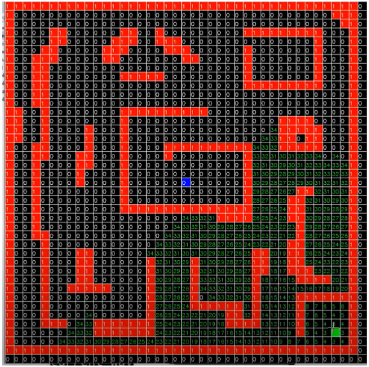
1	1	1	1	1	0	0	0	2	0	0
1	0	0	0	1	0	0	0	0	3	0
1	1	0	1	1	0	0	4	4	3	0
0	0	5	0	0	0	0	0	0	0	0
6	6	0	7	7	0	8	0	0	0	9
6	0	0	0	7	0	8	0	10	0	9
6	6	6	6	6	0	8	8	8	8	8

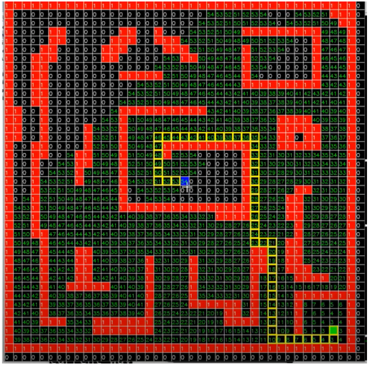
1	1	1	1	1	0	0	0	2	0	0
1	0	0	0	1	0	0	0	0	3	0
1	1	0	1	1	0	0	3	3	3	0
0	0	5	0	0	0	0	0	0	0	0
6	6	0	6	6	0	8	0	0	0	8
6	0	0	0	6	0	8	0	8	0	8
6	6	6	6	6	0	8	8	8	8	8

```
https://en.wikipedia.org/wiki/Connected-component_labeling
https://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm
http://aishack.in/tutorials/connected-component-labelling/
```

Luis - Option 1: Wavefront Expansion Algorithm

Implement for N robots, each with a specific start position and destination.

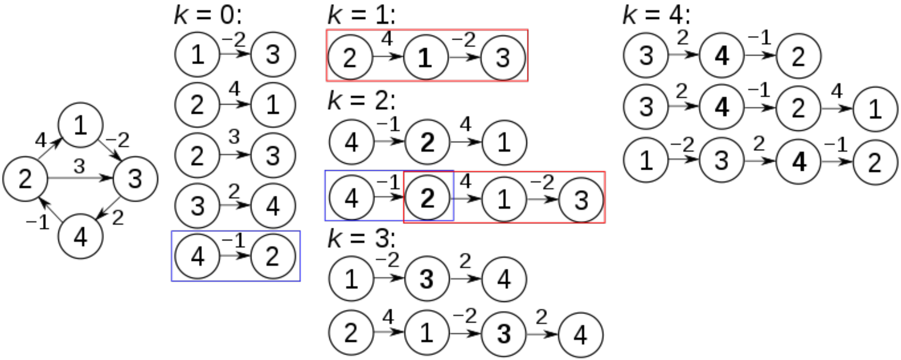




https://en.wikipedia.org/wiki/Wavefront_expansion_algorithm
https://www.youtube.com/watch?v=yeL7ICc8g4A

Luis - Option 2: Floyd's Algorithm

Shortest Paths Between All Pairs of Nodes



https://www-m9.ma.tum.de/graph-algorithms/spp-floyd-warshall/index_en.html
https://en.wikipedia.org/wiki/Floyd%E2%80%93Warshall_algorithm

Infrastructure: Shared-Memory Programming

For projects based on **Pthreads** and **OpenMP**, experiments will be run in the following server:

polifemo.tamps.cinvestav.mx (148.247.202.45)

You will use username "**cpar**" to connect to this server using **ssh**:

```
ssh cpar@polifemo.tamps.cinvestav.mx
```

Upload/download files to/from the server using commands **scp** or **rsync**.

Please **respect the times** assigned to each of you for using this server!

Infrastructure: Distributed-Memory Programming

For projects based on **MPI**, experiments will be run in **Cluster Neptuno**:

neptuno.tamps.cinvestav.mx (148.247.202.41)

You will use username "**cpar**" to connect to this server using **ssh**:

```
ssh cpar@neptuno.tamps.cinvestav.mx
```

Upload/download files to/from the server using commands **scp** or **rsync**.

Please **respect the times** assigned to each of you for using this server!