Projet Informatique

IN104

Natalia Díaz Rodríguez, ENSTA Paris, Institut Polytechnique Paris

Logistics

- Moodle Course page includes material, but
- The most up-to-date material is in the Course repository
- Project preferences: Make 3 choices (in decreasing order of preference) here
    - Deadline to choose one of the 8 topics: TOMORROW at NOON (1st come, 1st served)
    - Beware: may not be possible to accommodate everyone; we need balanced groups, respond timely today!

Logistics (II)

- IN104 must help you learn manage your autonomy and independence
    - Essential today for an engineer
    - 1 ECTS = 25h of work (all included) -> Expect Min. 2h weekly of extra work at home required
- What are normal grades from past years?
    - No Exam -> Everything between 10 -12 and 20.
    - Up to you to get beyond average
        - A self-learning project
        - Internet is to be used

Logistics (III): Evaluation

1. 25% **Source code**: features, tests, code documentation, etc. Source code must be documented, the report including a link to the repository must be in a .zip file. The repository will contain the sources, as well as a plain text file README.md which indicates the actual operational features and limitations. Python code should compile, the teaching assistants are not supposed to make significant corrections for it to compile: a code with few features, but that compiles and does not crash will be preferred to a more complete code but which is not directly operational!
2. 25% **Defense** (10 minutes presentation, 5 min. questions): The formal quality of the presentation will be an important element. The defense needs to include a demonstration on the basis of the source code, an analysis of the difficulties encountered and implemented solutions. It will not include a presentation of the problem or the method of resolution that the teaching assistant obviously knows already well. The defense is open to everyone (subject to the acceptance of the pair that will present). The chronological order of defense will be given by the list of each group.
3. 20% **Practice Analysis**: You will return a critical (max. 5 pages) report before the defense in which you analyze and criticize the progress of your project and its success and failure factors. This evaluation component also includes the oral treatment of this question during the defense.
4. 30% **Continuous Progress Evaluation** of the practical work during the practical lab sessions (based on Git commits).

Timetable

- Session 1 Projects presentations and Intro to GIT
    - TD: GIT Practical guide and exercises
- Session 2 Object Oriented Programming (OOP) and Python.
    - TD: Project 0: Unix/Python/ OOP Tutorial: Submit via Gradescope. Deadline: 1 week)
- Session 3 Test Driven Development (TDD).
    - TD: TDD Practical.
- Rest of sessions: Work on Project week
- Last Session Project Defense Day (10 min of soutenance per team including questions)

  Deadline to send report (containing link to project repository) to your Teaching Assistant: 2 days before defense day

Minimum to pass the course

- Implement and use some **classes** and **objects** in your project
- Implement and run successfully some **unit tests**.
- Play the game of push/pull:
    - Coordinate work in the team using **Git** to incrementally progress on the project
    - Your TA will check your Github repository *commit, pull, push*… through the course duration

Course expectations

- This is your (first) independent project course
  - i.e., autonomous work expected
  - Time in lab is not expected to be sufficient to hand in a high quality project and report
  - Expected total work hours: at the very least, same amount of lab time, at home
- **Report: 1 single PDF containing link to repository**:
  - Quality over Quantity
  - Results (plots, insights, conclusions and learnings):

more important than lengthy reports explaining the problem

If you feel lost

- Establish milestones with your partner
    - Do GIT code reviews (of each other, if needed)
- Force yourself: pose the right question you want to answer
- Your TA is there to support you!

If you feel lost

Mid-term report hand in:

- Show to your TA, at least, the first 50% of your project report in the equator of the course
  - makes sure you are on track
  - can help you not derail. You can include:
    - Project Requirements
    - Software Architecture
    - Classes, Tests, Plots, Versions, Documentation...

Course & Team Python Conventions

PEP-8: Defines Python coding practices https://www.python.org/dev/peps/pep-0008

- **Indentation**: 4 spaces (good editors will replace <TAB> by 4 spaces)
- **Variables**: `lower_case_with_underscores`
- **Functions**: `lower_case_with_underscores()`
- **Classes**: `UpperCamelCase`
- **Attributes**: `lower_case_with_underscores`
- **Protected attributes**: `_prefixed_with_1_underscore`
- **Constants**: `ALL_CAPS`
- **Modules**: `lowercase` (single word)

PEP-257: Documentation conventions.

- Prescribes the function or method's effect as a command ("Do this", "Return that:").
- *Docstring* should NOT be a "signature" (reiterating the function/method parameters, which can be obtained by instrospection).

References & Acknowledgements

- Jean-Didier Garaud. Coding Best Practices
- Resources to learn Scientific Python: scipy, pandas, numpy: https://github.com/paris-saclay-cds/data-science-workshop-2019
- Lutz, M CC. (2011). Programming Python (4.a ed.). EE. UU.: O’Reilly Media.
- Lutz, M (2013). Learning Python (5.a ed.). EE. UU.: O’Reilly Media.
- Martelli, A (2017). Python in a Nutshell (3.a ed.). EE. UU.: O’Reilly Media.
- Pilgrim, M (2011). Dive Into Python 3(online). APress. http://www.diveintopython3.net
- Python Software Foundation. Python Official Webpage. <http://www.python.org>
- González Duque, R. Python para todos (on line).
- Photography: Sandy Skoglund

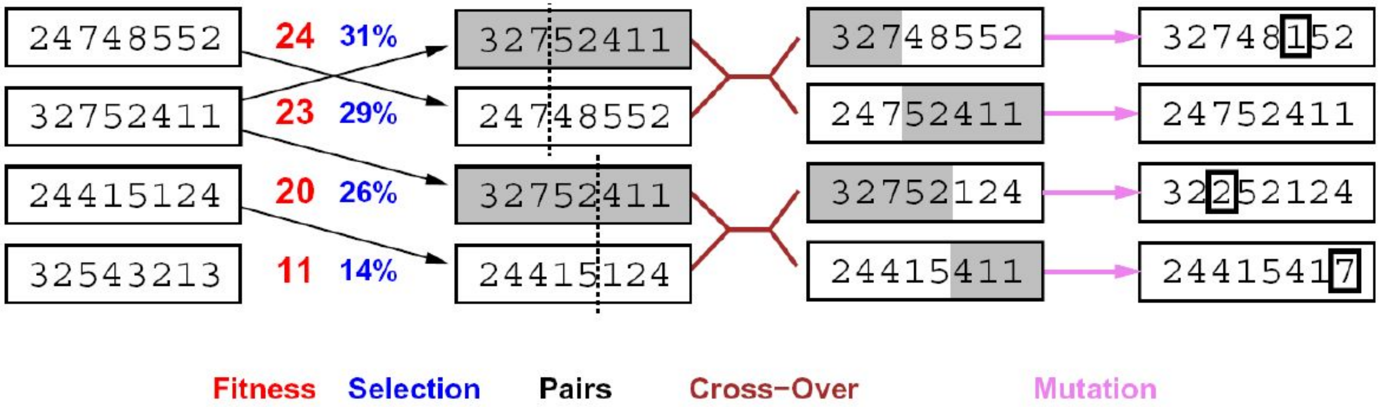
5 min Project Pitchs Time!

Project 1: GENETIC ALGORITHMS

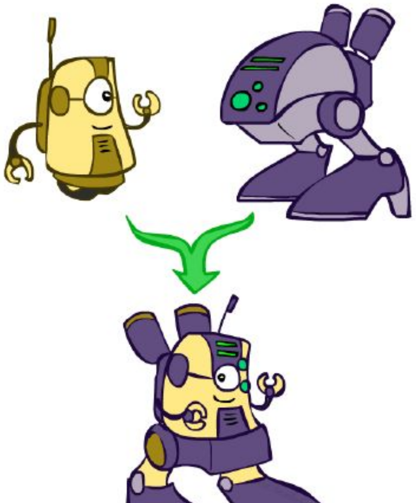
Natalia Díaz Rodríguez

Github username: NataliaDiaz

- Commonly used to generate high-quality solutions to **optimization and search** problems by relying on bio-inspired **operators**
- A **metaheuristic** belonging to the larger class of evolutionary algorithms (EA).
  - Inspired by a natural selection metaphor:
    - Keep best N hypotheses at each step (**selection**) based on a fitness function.
    - Have a pairwise **crossover** operator, with optional **mutation** to give variety



	Fitness	Selection	Pairs	Cross-Over	Mutation
24748552	24	31%	32752411	32748552	32748152
32752411	23	29%	24748552	24752411	24752411
24415124	20	26%	32752411	32752124	32252124
32543213	11	14%	24415124	24415411	24415417

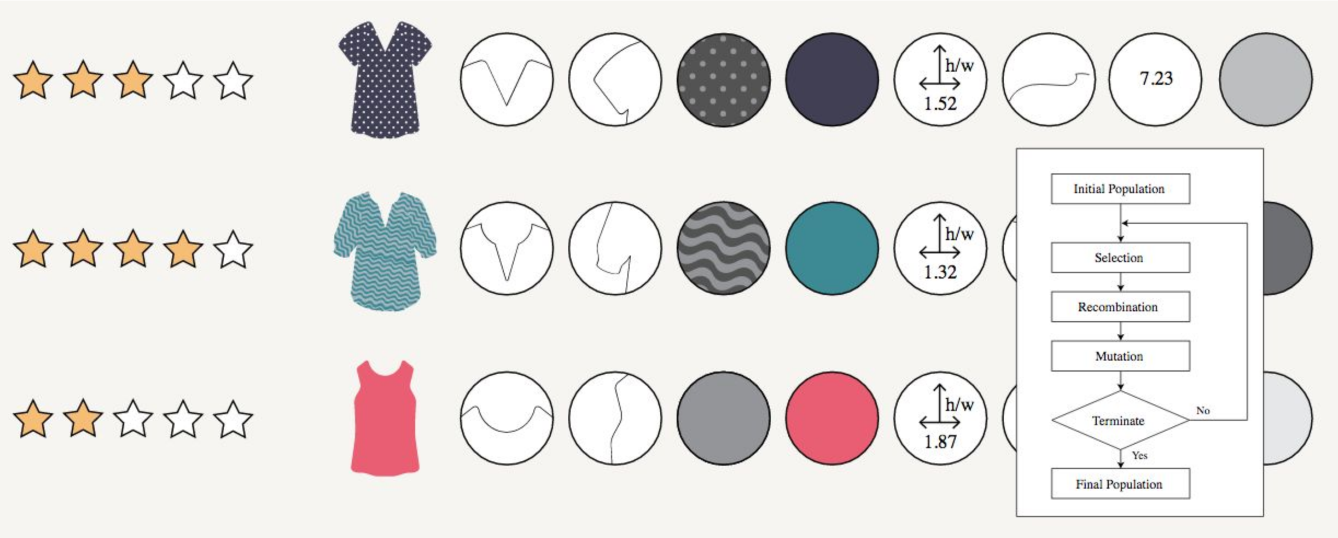


Project 1: GENETIC ALGORITHMS

Natalia Díaz Rodríguez

A Visual Guide to Evolution Strategies

OCTOBER 29, 2017



Initial Population

Selection

Recombination

Mutation

Terminate

No

Yes

Final Population

Project 1: GENETIC ALGORITHMS

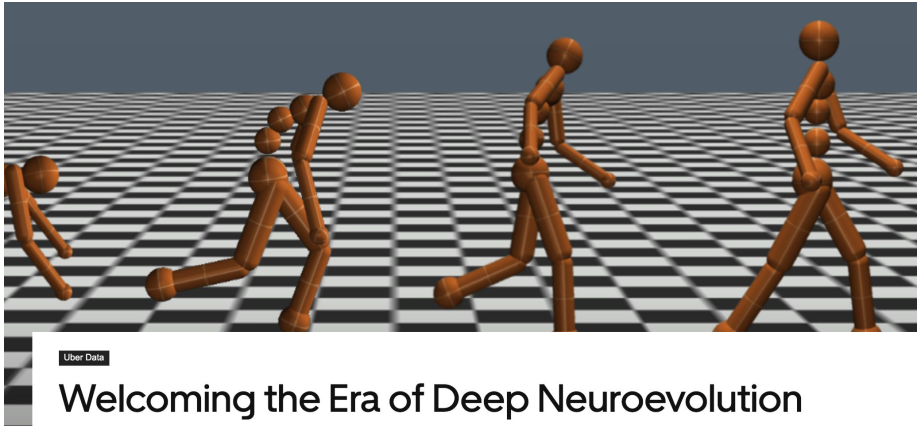
Natalia Díaz Rodríguez

Intro reads:

- GA: https://en.wikipedia.org/wiki/Genetic_algorithm
- EA: https://en.wikipedia.org/wiki/Evolutionary_algorithm
- Visual Evolution Strategies: http://blog.otoro.net/2017/10/29/visual-evolution-strategies/
- The Era of Deep Neuroevolution (Uber AI Labs): https://eng.uber.com/deep-neuroevolution/
- An intro to genetic algorithms and an example of real industry application on data-driven fashion design https://multithreaded.stitchfix.com/blog/2016/07/14/data-driven-fashion-design/

Contact: natalia.diaz(at)ensta-paris(dot)fr

https://github.com/NataliaDiaz/IN_104-Projet-Informatique



Uber Data

Welcoming the Era of Deep Neuroevolution

Project 2: FAKE NEWS DETECTOR

Nermine ALI

- Why Fake News Detector?
  - News spread quickly on social media: True or Fake?
  - Fake news are misleading…

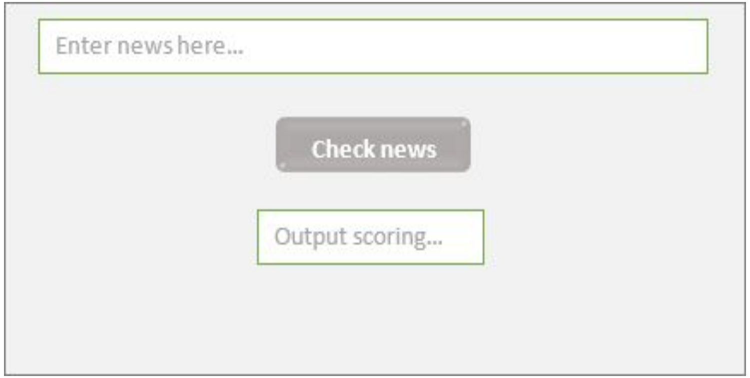
- Task
  - Create a Fake News Detector using AI

https://drive.google.com/file/d/1yPo8qMJtRc7MiXpWQ84251X07UMVqbZa/view?usp=sharing

Project 2: FAKE NEWS DETECTOR

Nermine ALI

- Python as Programming language
- Expected outcome
  - Web page: check button + textbox
  - Input: news article (URL)
  - Output: Score (0 to 100) for the credibility of the news
- Steps
  1. Availability in trusted news sources
     Scoring according to the number of mentions in news sources (35%)
  2. AI Scoring (65%)



Project 2: FAKE NEWS DETECTOR

Nermine ALI

- How to implement
  - Get news from social media, websites…
  - Look for it on trusted sources
  - Ranking for each news source (based on a trained neural network)
  - Credibility of suspicious news
- References
  - https://www.sciencedirect.com/science/article/pii/S0378437119317546
- Contact: nermineali.lb@gmail.com
- More details : https://github.com/nermine-ali/Fake-news-detector

Project 3: Robotics at home, turn your phone into an IMU ! #stayhome

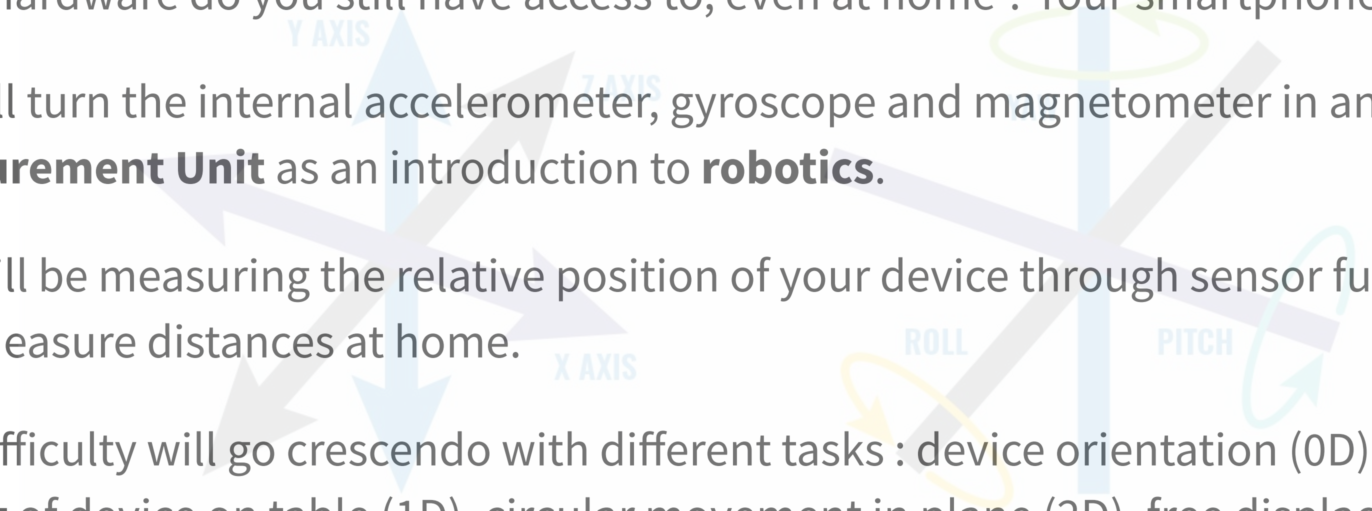
Victor Talpaert

What hardware do you still have access to, even at home ? Your smartphone !

We will turn the internal accelerometer, gyroscope and magnetometer in an **Inertial Measurement Unit** as an introduction to **robotics**.

You will be measuring the relative position of your device through sensor fusion and use it to measure distances at home.

The difficulty will go crescendo with different tasks : device orientation (0D), rectiligne sliding of device on table (1D), circular movement in plane (2D), free displacement (3D).



victor.talpaert@ensta-paris.fr, U2IS bureau R.2.11

Project 3: Robotics at home, turn your phone into an IMU ! #stayhome

Victor Talpaert

Prerequisite : go to https://whatwebcando.today/device-motion.html on your device and verify at the bottom if motions are detected (sorry for gatekeeping, but you need a smartphone and computer at home for this project).

Develop with your team using a repo, write code in python

HTML 5 WebSockets

Run provided server code on computer, open a webpage with your device through local WiFi network to start streaming data back to the computer.

Start code will be here with instructions : https://github.com/vtalpaert/phone-imu

Project 3: Robotics at home, turn your phone into an IMU ! #stayhome

Victor Talpaert

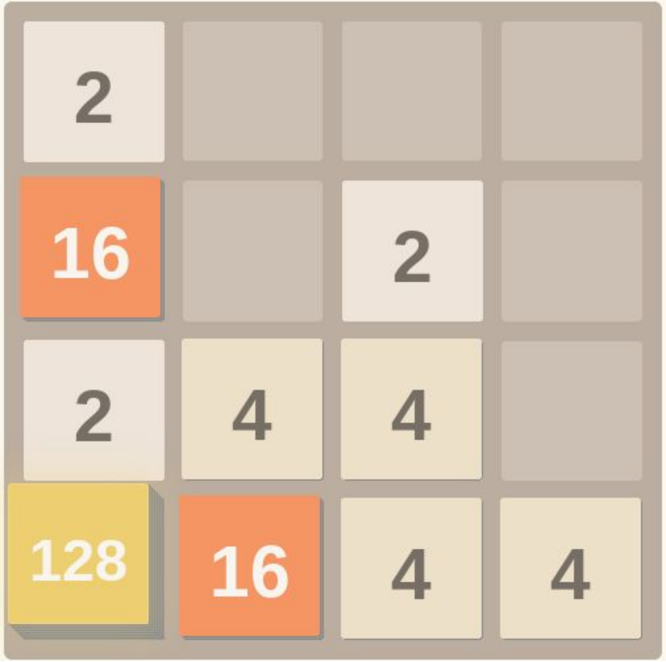
- I expect you to try both state of the art sensor fusion, but also your own algorithms
- Explain WHY things work or not (such as double integration drift)
- Compete with others on the different tasks, share results on the Telegram Channel
- Visualisation is an important part of this project, in the report and in live demo
- Eventual Pull Requests for the web page rendering to my repo are also graded
- Code not written by your team is not part of your grade
- No “but it worked on my machine”

You can see this project as transforming your device in an IoT sensor, or as making a ruler app, have fun !

Project 4: 2048

Olivier Mullier (contact: mullier@ensta-paris.fr)

The goal is to implement the “2048” game from G. Cirulli.



2			
16		2	
2	4	4	
128	16	4	4

You will have to produce:

- An implementation (in C or C++) of the game;
- A graphical user interface (GUI) to play it;
- An AI able to play the game.

Further details here:

https://drive.google.com/file/d/1GGMIpgft5H2Xn4TrV_Vnw1bAz2gDmgoR/view?usp=sharing

Project 5: Trading Bot in the European Weather and Energy Markets

Eliot Tabet

- Compréhension du marché de l’énergie (gaz/électricité) et l’impact du climat dans ce marché
- Prévision de la consommation
- Prise de décision sur la position à prendre (Buy/Sell) en se basant sur les algorithmes de machine learning

https://drive.google.com/drive/folders/1o9m_fSyRIx9L26HXHe1cnxhT_F9vnxzB?usp=sharing

eliot.tabet@polytechnique.edu

Project 5: Trading Bot in the European Weather and Energy Markets

Eliot Tabet

- Python : Pandas, Numpy, SciPy, SkLearn, Matplotlib, dfply
- Supervised Learning Algorithms (Classification and Regression): Random Forest, Logistic Regression, Non-linear regression, Multidimensional Linear Regression

https://drive.google.com/drive/folders/1o9m_fSyRIx9L26HXHe1cnxhT_F9vnxzB?usp=sharing

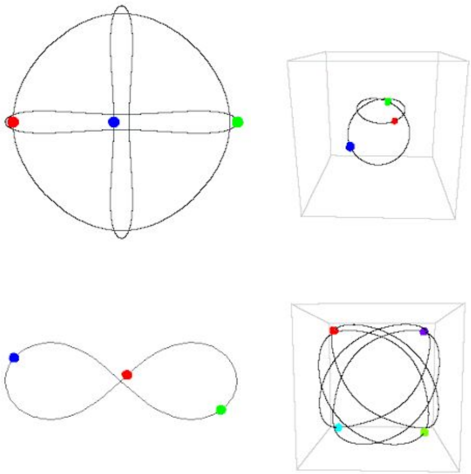
eliot.tabet@polytechnique.edu

Ismail Bennani <ismail.lahkim.bennani@ens.fr>

Project 6: N-Body problem

Simulate a (very) tiny universe

- “The n-body problem is the problem of predicting the individual motions of a group of celestial objects interacting with each other gravitationally” - Wikipedia



GIFs from http://rebloggy.com/post/gif-gifset-science-math-physics-n-body-problem-orbits/39788206329

Why is it difficult ?

- Computing forces:
    - ○ Naive method: compute the N^2 gravitational forces → not effective, struggles with >100 bodies and euler integration on my computer
- Integrating the system: that is, computing the dynamics
    - ○ Naive method: explicit euler integration → very inaccurate, the system diverges quickly

Examples: https://vanderbei.princeton.edu/WebGL/nBody.html

Ismail Bennani <ismail.lahkim.bennani@ens.fr>

Project 6: N-Body problem

Simulate a (very) tiny universe

Your job (mandatory):

- Write a physics engine computing the forces applied to each body
- Write a an ODE solver to integrate the dynamics
- Write a graphical interface to show the results

Tool suggestions	
Language	Python
2D drawing lib	Pygame, PySFML
3D drawing lib	PyOpenGL

The rest is up to you, you can:

- Compute physics faster, e.g. simulate an asteroid belt (700.000 to 1.7M asteroids in our solar system)
- Compute more accurate dynamics (Leapfrog algorithm, Runge-Kutta family, …)
- Handle collisions between bodies
- Find stable initial positions automatically
- Improve the GUI, e.g. add bodies with a click, add a camera, draw path of bodies, ...
- … be creative !

More details: https://github.com/ismailbennani/IN104/blob/master/sujet/sujet.pdf

Git repo: https://github.com/ismailbennani/IN104

Github username: ismailbennani

Project 7: Artificial Intelligence for 2 players games

Clement Masson

You will learn the basics of algorithms used for most 2 players board games.

(disclaimer: this is NOT a Machine Learning project. It would require too much time to tackle the ML part)

masson.cle@gmail.com

Project 7: Artificial Intelligence for 2 players games

Clement Masson

Language: python

Expected work: project will start with many guidelines so you know where to start.

Evaluation:

- Oral presentation & report : 50 %
- AI Competition ranking : 25 %
- Code quality, good use of git, deadline respect, work in group … : 25 %

Useful links:

Project website (forum, guidelines, help, …): https://sites.google.com/view/ensta-in104

Wiki : https://github.com/clement-masson/aiarena/wiki

masson.cle@gmail.com

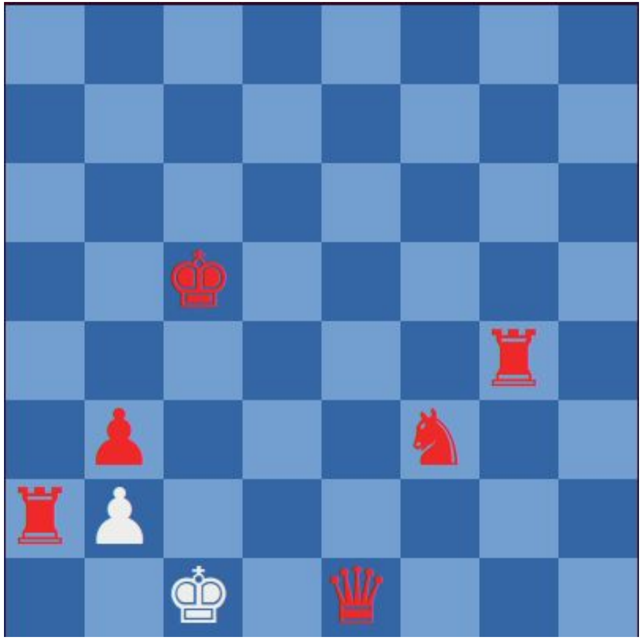
Project 7: Artificial Intelligence for 2 players games

Fun :

- You can play against your own AI
- Your AI will compete against other groups' AI !

Interesting theory aspects :

- Recursivity, Trees
- Minimax, alpha-beta pruning
- Object Programming
- Code profiling and optimization

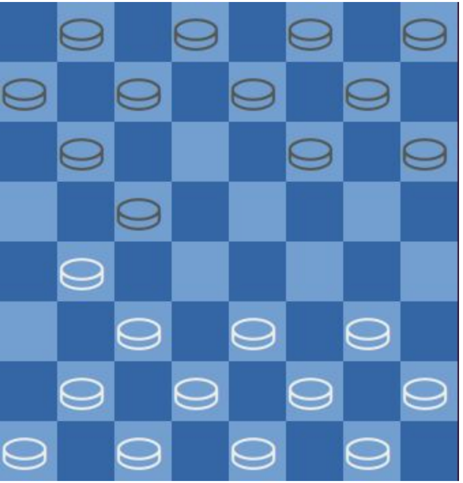


masson.cle@gmail.com

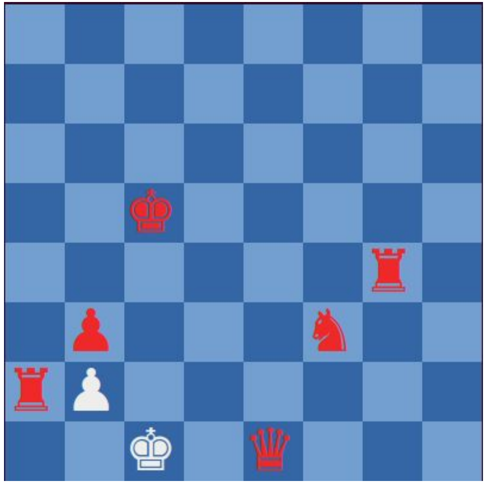
Project 7: Artificial Intelligence for 2 players games

Clement Masson

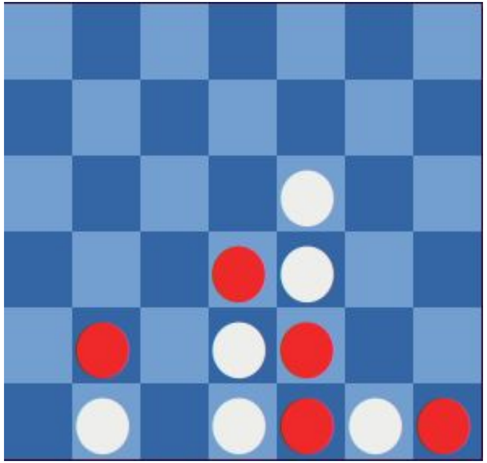
Some code is already provided to let you quickly start your implementation of AIs and play against them !



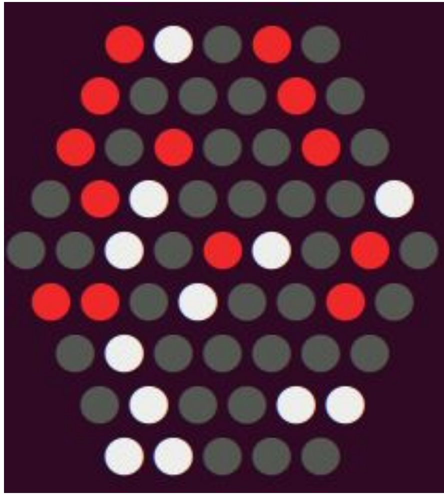
checkers



chess



connect4



abalone

masson.cle@gmail.com

Project 8: Hanabi AI

Jean-Didier Garaud (jean-didier.garaud@onera.fr)

A **cooperative** game (2-5 players)

Spiel des Jahres 2013

Very simple to explain, very tricky to play well

Challenging for AI design:

- Imperfect information
- AI has to cooperate with human
- Coordination, trust, empathy

Currently a research subject (google summer, deepmind, ...)

Project 8: Hanabi AI

How to play

The goal is simply to play all 25 cards (1 to 5 in each color) in the correct order.

BUT : You don't see your cards,
You only see your partners'!

Each turn you must do 1 (and only 1) of:

- Give a clue
- Play a card
- Discard a card

Project 8: Hanabi AI -- Objectives

OO Programming with Python 3:

use an existing library and extend it:

`https://github.com/JDGaraudEnsta/hanabi`

Task: develop one or more AI’s, based on:

- deterministic strategies
- human conventions & strategies
- or academic papers (hat strategy, deep learning)

Let it play against (with?) itself or a human player

(optionally) Create a Graphical Interface

```
[jd@ananke hanabi]$ hanabi
Let's start a new game
Here are the starting hands:
[Y4 Y3 R4 B2 Y2, R2 Y1 R1 G1 W4]

Alice this is what you remember:   ** ** ** ** **
        this is what you see:        R2 Y1 R1 G1 W4
What do you want to play?
        (d)iscard a card (12345)
        give a (c)lue (RBGWY 12345)
        (p)lay a card (12345)
        e(x)amine the piles
hanabi> c1
Alice gives a clue: 1

Benji this is what you remember: ** *1 *1 *1 **
        this is what you see:      Y4 Y3 R4 B2 Y2
What do you want to play?
```

Project 8: Hanabi AI

Online resources:

[BGA](https://fr.boardgamearena.com/#!gamepanel?game=hanabi)

[Hanabi strategies](https://github.com/Zamiell/hanabi-conventions)

[HanSim: the hat guessing strategy]
(https://d0474d97-a-62cb3a1a-s-sites.googlegroups.com/site/rmgpgrwc/research-papers/Hanabi_final.pdf)

[deepmind Hanabi] (https://arxiv.org/abs/1902.00506)

[facebook’s hanabi]
(https://ai.facebook.com/blog/building-ai-that-can-master-complex-cooperative-games-with-hidden-information/)

Carlos Garcia-Guillamon
carlos.garcia-guillamon@safrangroup.com
carlosgguil@outlook.es
Github: carlosgguil

Project 9: Rocket design with python

Use Python to create a tool for designing and simulating rockets

- Choose one (or several) mission(s)
- Develop a GUI (tkinter) for rocket design
- Construct a rocket database with the aid of the GUI
  - Handling classes
  - Data exploration (pandas)
- Simulate the mission
  - Plot rocket performance (matplotlib, pygame)
  - Plot the animated trajectory

All info in github repository: https://github.com/carlosgguil/IN_104-project_09-rocket_design

You must create your own repository and commit on it

Carlos Garcia-Guillamon
carlos.garcia-guillamon@safrangroup.com
carlosgguil@outlook.es
Github: carlosgguil

Project 9: Rocket design with python

If you want to go further, you can:

- Add more complexity by considering multistage rockets
- Check different trajectories
- Implement a re-entry trajectory

Evaluation:

- 40 % source code
- 25 % continuous evaluation
- 25 % final presentation
- 10 % final report

Project 10: Build a navigation app for your phone*

Stephen Creff

*the smartphone is not provided

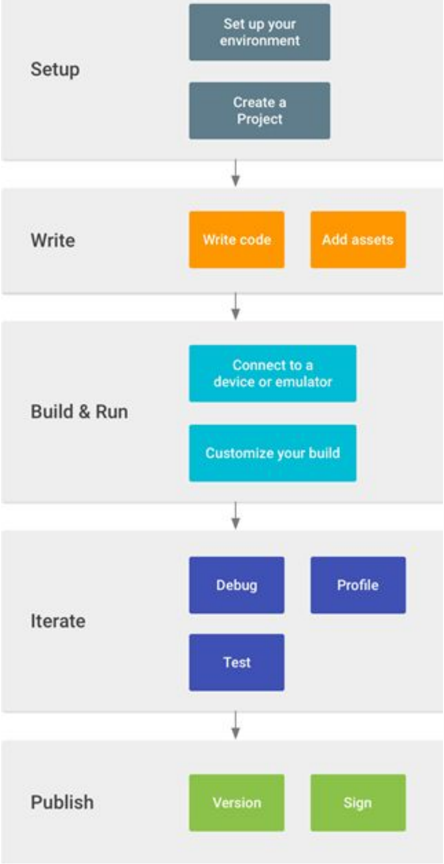
- Use native or cross-platform app development framework to build an iOS or Android app,
- Choose your language (Python, Java, …) and find your development framework (tech stack),
- Understand the framework and the APIs (widgets, …)

- Make your System Analysis and Requirements,make it incremental!
  - Add features and sensors solicitation step by step, from trivial to rich location-based
  - Define map (simplified, real, feature-rich), set geo-markers, calculate and draw route, …
- Define a storyboard and mock-up the Interface design
- Define and build functions using smartphone sensors
  - Touch screen, Geolocation service, …
- Package & deploy your app on the targeted OS

stephen.creff@gmail.com

Project 10: Build a navigation app for your phone*

Stephen Creff



Setup: Set up your environment → Create a Project

Write: Write code, Add assets

Build & Run: Connect to a device or emulator → Customize your build

Iterate: Debug, Profile, Test

Publish: Version, Sign

Key points of the project:

- Follow a full system development workflow,
- Learn to program and use dedicated APIs,
- Routing problem, pathfinding algorithms,
- *And, have fun with your phone !*

Some frameworks to build a Mobile Applications:

- A python framework : https://kivy.org/#home
- A java framework : https://www.codenameone.com/
- Other framework, with combination of many languages i.e., HTML5, JavaScript, and CSS and Cordova wrapper : https://ionicframework.com/

*the smartphone is not provided

Link to slides and course material:
https://github.com/NataliaDiaz/IN_104-Projet-Informatique

Lecture 1: Intro to GIT

Lecture 2: Intro to Object Oriented Programming (OOP)

Lecture 3: Test Driven Development (TDD) in Python