IMMUNE
TECHNOLOGY INSTITUTE

COMPUTER ENTREPRENEURSHIP BACHELOR

BLOCK NAME	**ARTIFICIAL INTELLIGENCE**
BLOCK CODE	**CS-L6B1**

COURSE	**3**
LEVEL	**6**
CREDITS	**2**
CLASS HOURS	**20**
HOMEWORK	**30**
TOTAL HOURS	**50**

DESCRIPTION

This block introduces basic artificial intelligence techniques except machine learning, which has been reserved for the Data Science block.
We face the challenge of creating a software for a 'new device' (simulating we are in 1986) which will make use of GPS technology to guide car drivers to its desired destination. Our department is in charge of the algorithms that will allow such device choose the best route from a start location to the destination.

PRE-REQUISITES

Programming skills are needed.
CS-L1B1, CS-L1B2

OBJECTIVES

The goal is for students to understand the basic artificial intelligence principles and techniques.

SKILLS TO BE DEVELOPED

1 - Logic problems solving.
1.1 - Understand how logic problems are formulated and how to test if a given solution is correct.
1.2 - Design and develop algorithms that solve logic problems.
2 - Search strategies.
2.1 - Understand how search problems are formulated and how to test if a given solution is correct.
2.1 - Design and develop algorithms able to search a solution inside a solutions space.
3 - Expert systems.
3.1 - Understand how expert systems work.
3.2 - Design and develop an expert system.
4 - Neural networks.
4.1 - Understand the principles of artificial neural networks and how they are first trained and then used to infer results.
4.2 - Design and develop a neural network.
5 - Global lessons about Artificial Intelligence.
5.1 - Choose the correct kind of artificial intelligence algorithm to solve a certain problem.

SYLLABUS

1 - Logic problem solving.
2 - Search strategies.
3 - Expert systems.
4 - Neural networks.

METHODOLOGY

Resolution of practical activities supervised by the mentor. Compulsory attendance.

DEDICATION AND EVALUATION

The student must pass the mandatory activities (challenges/projects) that are covered in the block.
Each challenge/project produces its own score and has been designed to cover certain block percentages.
Such score is 80% objective (the program that solves the challenge/project works without errors and producing the expected results) and 20% subjective (solution elegance, how clean the code is, documentation).
Block scores are finally calculated by prorating individual activities with respect to their block coverage percentages.