

Scoring Indicators

COURSENAME: Fundamentals of Artificial Intelligence and Machine Learning

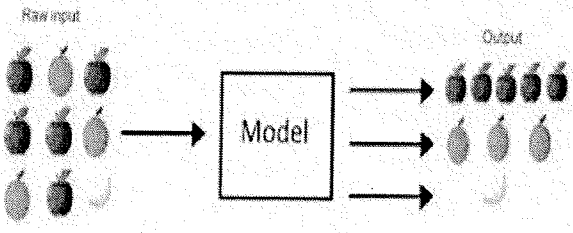
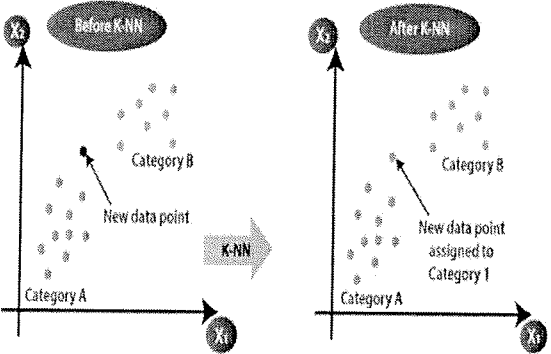
COURSECODE:5133C

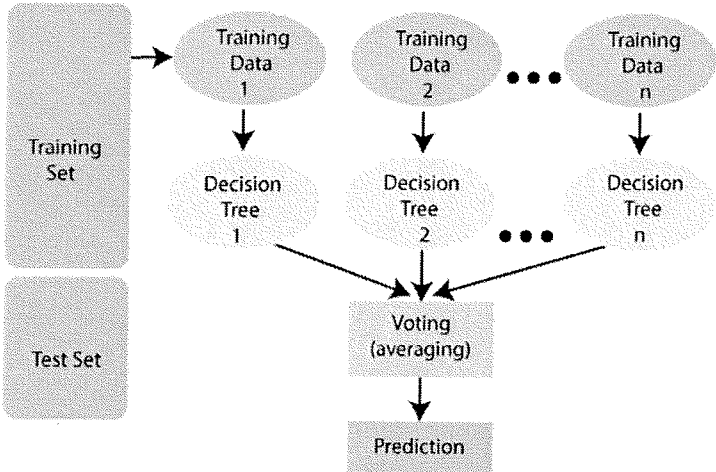
QID: 2109230300A

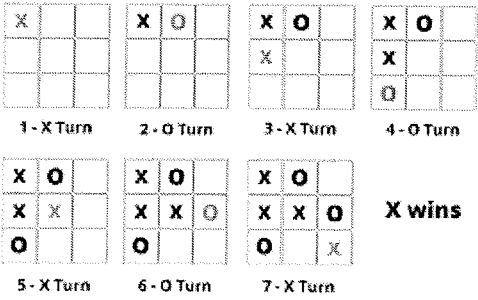
QNo	Scoring Indicators	Splits core	SubTotal	Totals core
<u>PART A</u>				9
I.1	Artificial Intelligence is a method of making a computer, a computer-controlled robot, or a software think intelligently like the human mind	1	1	9
I.2	Supervised learning	1	1	
I.3	True	1	1	
I.4	Key:value	1	1	
I.5	Classification,Regression	1	1	
I.6	Support Vector Machine	1	1	
I.7	Unsupervised	1	1	
I.8	Reinforcement learning	1	1	
I.9	Minimax	1	1	
<u>PART B</u>				24
II 1	<p>Necessity of learning Artificial intelligence</p> <ul style="list-style-type: none"> ☉ Handle large amounts of data in an efficient way. With the advent of CloudComputing, we are now able to store huge amounts of data. ☉ Ingest data simultaneously from multiple sources without any lag. ☉ Index and organize data in a way that allows us to derive insights. ☉ Learn from new data and update constantly using the right learning algorithms. ☉ Think and respond to situations based on the conditions 	Any 3 points-1 marks each	3	3

	<p>in real time.</p> <ul style="list-style-type: none"> ⦿ Continue with tasks without getting tired or needing breaks. ⦿ AI techniques are actively being used to make existing machines smarter 			
II 2	<p style="text-align: center;">Various artificial intelligence tools</p> <p>Scikit Learn <u>Scikit-learn</u> is one of the most well-known ML libraries. It underpins many administered and unsupervised learning calculations.</p> <p>Tensorflow The fascinating thing about <u>Tensorflow</u> is that when you compose a program in Python, you can arrange and keep running on either your CPU or GPU.</p> <p>Keras If you like the Python-way of doing things, Keras is for you. It is a high-level library for neural networks</p> <p>PyTorch <u>PyTorch</u> is an AI system created by Facebook. Its code is accessible on <u>GitHub</u></p> <p>Theano</p> <ul style="list-style-type: none"> • It was created to make actualizing profound learning models as quick and simple as feasible for innovative work. 	Any 3-1 mark each	3	3
II. 3	<p>Write a python program to find largest of two numbers</p> <pre>print("Enter Two Numbers: ",) numOne = int(input()) numTwo = int(input()) if numOne>numTwo: print("\nLargest Number =", numOne) else: print("\nLargest Number =", numTwo)</pre>	Declaratio n-1 + logic-2	3	3

II. 4	<p>List in python with example</p> <ul style="list-style-type: none"> • Ordered sequence of values • Written as a sequence of comma-separated values between square brackets • Values can be of different types <ul style="list-style-type: none"> – usually the items all have the same type <pre>>>> lst = [1, 2, 3, 4, 5] >>> lst [1, 2, 3, 4, 5] >>> type(lst) <type 'list'></pre>	Definition -1 + Syntax-2	3	3
II. 5	<p style="text-align: center;">Exception handling in python</p> <p>In Python, there are several built-in exceptions that can be raised when an error occurs during the execution of a program.</p> <p>SyntaxError: This exception is raised when the interpreter encounters a syntax error in the code, such as a misspelled keyword, a missing colon, or an unbalanced parenthesis.</p> <p>TypeError: This exception is raised when an operation or function is applied to an object of the wrong type, such as adding a string to an integer.</p> <p>NameError: This exception is raised when a variable or function name is not found in the current scope.</p> <p>IndexError: This exception is raised when an index is out of range for a list, tuple, or other sequence types.</p> <p>KeyError: This exception is raised when a key is not found in a dictionary.</p> <p>ValueError: This exception is raised when a function or method is called with an invalid argument or input, such as trying to convert a string to an integer when the string does not represent a valid integer.</p> <p>AttributeError: This exception is raised when an attribute or method is not found on an object, such as trying to access a non-existent attribute of a class instance.</p> <p>IOError: This exception is raised when an I/O operation, such as reading or writing a file, fails due to an input/output error.</p>	Any 3-1 mark each	3	3
II. 6	<p>Classification</p> <ul style="list-style-type: none"> ▶ Classification algorithms are used when the output variable is categorical, which means there are two classes such as Yes-No, Male-Female, True-false, etc. ▶ Random Forest ▶ Decision Trees ▶ Logistic Regression ▶ Support vector Machines 	Definition -1.5 + Figure-1.5	3	3

				
<p>II. 7</p>	<p>K-Nearest Neighbour(KNN)</p> <ul style="list-style-type: none"> ▶ K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. ▶ K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. <p>K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm</p>	<p>1.5+1.5</p>	<p>3</p>	<p>3</p>
				

<p>II. 8</p>	<p>Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of <i>combining multiple classifiers to solve a complex problem and to improve the performance of the model.</i></p> <p>As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."</p> <p>The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.</p> 	<p>1.5+1.5</p>	<p>3</p>	<p>3</p>
<p>II. 9</p>	<p>Describe Combinatorial Search Find atleast one example of a combinatorial structures of a particular type (if one exists). Combinatorial search works efficiently by using pruning strategies. These strategies avoid testing all possible solutions by eliminating the ones that are obviously wrong. This helps to save time and effort.</p>	<p>3</p>	<p>3</p>	<p>3</p>
	<p>Tic tac toe game play_game() is the main function, which performs the following tasks :</p> <ul style="list-style-type: none"> · Calls create_board() to create a 3×3 board and initializes with 0. · For each player (1 or 2), calls the random_place() function to randomly choose a location on board and mark that location with the player number, alternatively. · Print the board after each move. · Evaluate the board after each move to check whether a row or column or diagonal has the same player number. 			

II. 10	<p>If so, displays the winner's name. If after 9 moves, there is no winner then displays -1.</p> <p style="text-align: center;">Tic Tac Toe - Gameplay</p>  <p>1 - X Turn 2 - O Turn 3 - X Turn 4 - O Turn</p> <p>5 - X Turn 6 - O Turn 7 - X Turn</p> <p style="text-align: right;">X wins</p>	3	3	3
<u>PART C</u>				42
III	<p>Different types of learning in AI</p> <ul style="list-style-type: none"> ⊙ Supervised Learning ⊙ Unsupervised Learning ⊙ Reinforcement Learning <p>Three types of Learning</p> <p>Supervised Learning: The machine has a "teacher" who guides it by providing sample inputs along with the desired output. The machine then maps the inputs and the outputs. This is similar to how we teach very young children with picture books.</p> <p>Unsupervised means to act without anyone's supervision or direction. unsupervised learning, the model is given a dataset which is neither labelled nor classified. The model explores the data and draws inferences from datasets to define hidden structures from unlabelled data.</p> <p>Reinforcement Learning (RL) is a sub-field of Machine Learning where the aim is create agents that learn how to operate optimally in a partially random environment by directly interacting with it and observing the consequences of its actions.</p>	1+2+2+2	7	7

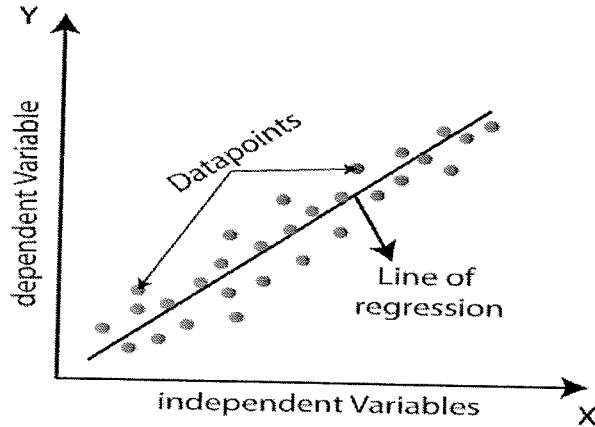
IV	<p>Different fields of artificial intelligence</p> <ul style="list-style-type: none"> • Logic-based AI: Mathematical logic is used to execute computer programs in logic-based AI. A program written in logic-based AI is basically a set of statements in logical form that expresses facts and rules about a problem domain. This is used extensively in pattern matching, language parsing, semantic analysis, and so on. • Search: Search techniques are used extensively in AI programs. These programs examine many possibilities and then pick the most optimal path. For example, this is used a lot in strategy games such as chess, networking, resource allocation, scheduling, and so on. • Knowledge representation: The facts about the world around us need to be represented in some way for a system to make sense of them. The languages of mathematical logic are frequently used here. If knowledge is represented efficiently, systems can be smarter and more intelligent. Ontology is a closely related field of study that deals with the kinds of objects that exist. • Planning: This field deals with optimal planning that gives us maximum returns with minimal costs. These software programs start with facts about the situation and a statement of a goal. These programs are also aware of the facts of the world, so that they know what the rules are. From this information, they generate the most optimal plan to achieve the goal. • Heuristics: A heuristic is a technique used to solve a given problem that's practical and useful in solving the problem in the short term, but not guaranteed to be optimal. This is more like an educated guess on what approach we should take to solve a problem. In AI, we frequently encounter situations where we cannot check every single possibility to pick the best option. Thus, we need to use heuristics to achieve the goal. They are used extensively in AI in fields such as robotics, search engines, and so on. • Genetic programming: Genetic programming is a way to get programs to solve a task by mating programs and selecting the fittest. The programs are encoded as a set of genes, using an algorithm to get a program that can perform the given task well. 	4+3	7	7
----	--	-----	---	---

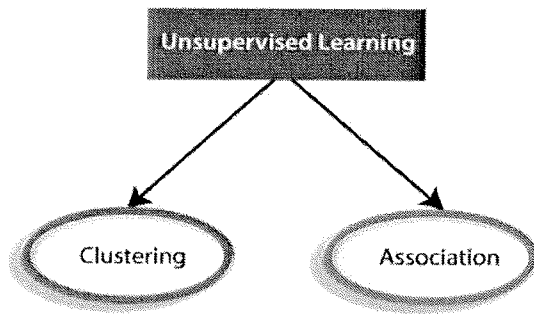
	<ul style="list-style-type: none"> • Machine learning and pattern recognition: This is perhaps the most popular form of AI out there. We design and develop software that can learn from data. Based on these learning models, we perform predictions on unknown data. One of the main constraints here is that these programs are limited to the power of the data. 			
V	<p>Creating a Function</p> <p>In Python a function is defined using the def keyword:</p> <p>Example</p> <pre>def my_function(): print("Hello from a function")</pre> <p>Calling a Function</p> <p>To call a function, use the function name followed by parenthesis:</p> <pre>def my_function(): print("Hello from a function")</pre> <p>my_function()</p> <p>Arguments</p> <p>Information can be passed into functions as arguments.</p> <p>Arguments are specified after the function name, inside the parentheses. You can add as many arguments as you want, just separate them with a comma.</p> <pre>def my_function(fname): print(fname + " Refsnes")</pre> <pre>my_function("Emil") my_function("Tobias") my_function("Linus")</pre>	3+2+2	7	7

	<p>control statements with the help of example</p> <p>Break</p> <p>In Python, Loops are used to iterate repeatedly over a block of code. In order to change the way a loop is executed from its usual behavior, control statements are used.</p> <p>Break statement</p> <p>The break statement in Python is used to terminate or abandon the loop containing the statement and brings the control out of the loop.</p> <pre> age = "\n Please enter your age: " while True: age = input if age >= 18: break else: print ("You're not eligible to vote") </pre>			
VI	<p>Continue statement</p> <p>When a program encounters a continue statement in Python, it skips the execution of the current iteration when the condition is met and lets the loop continue to move to the next iteration</p> <pre> for letter in 'Flexiple': if letter == ' ': continue print ('Letters:', letter) </pre> <p>Pass statement</p> <p>The pass statement is a null operator and is used when the programmer wants to do nothing when the condition is satisfied.</p> <pre> for letter in 'Flexiple': if letter == 'x': pass print ('Letters:', letter) </pre>	3+2+2		7

VII	<p>Polymorphism is often used in Class methods, where we can have multiple classes with the same method name.</p> <p>Example</p> <p>Different classes with the same method:</p> <pre> class Car: def __init__(self, brand, model): self.brand = brand self.model = model def move(self): print("Drive!") class Boat: def __init__(self, brand, model): self.brand = brand self.model = model def move(self): print("Sail!") class Plane: def __init__(self, brand, model): self.brand = brand self.model = model def move(self): print("Fly!") car1 = Car("Ford", "Mustang") #Create a Car class boat1 = Boat("Ibiza", "Touring 20") #Create a Boat class plane1 = Plane("Boeing", "747") #Create a Plane class for x in (car1, boat1, plane1): x.move() </pre>	3+4	7	7

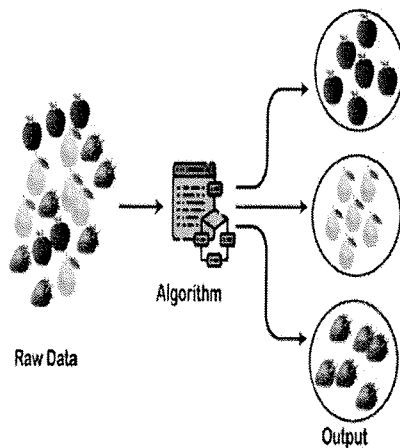
VIII	<p>Looping statements with the help of example</p> <p>While Loop in Python In python, a <u>while loop</u> is used to execute a block of statements repeatedly until a given condition is satisfied. And when the condition becomes false, the line immediately after the loop in the program is executed.</p> <pre>count = 0 while (count < 3): count = count + 1 print("Hello Geek")</pre> <p>For Loop in Python <u>For loops</u> are used for sequential traversal. For example: traversing a list or <u>string</u> or array etc. In Python, there is “for in” loop which is similar to <u>for each</u> loop in other languages. Let us learn how to use for in loop for sequential traversals.</p> <pre>n = 4 for i in range(0, n): print(i)</pre>	4+3	7	7
IX	<p>Linear regression</p> <ul style="list-style-type: none"> ▶ It is a statistical method that is used for predictive analysis ▶ Linear regression is used in many different fields, including finance, economics, and psychology, to understand and predict the behavior of a particular variable. For example, in finance, linear regression might be used to understand the relationship between a company’s stock price and its earnings or to predict the future value of a currency based on its past performance. 			

	 <p>dependent Variable</p> <p>Y</p> <p>X</p> <p>independent Variables</p> <p>Datapoints</p> <p>Line of regression</p> <ul style="list-style-type: none"> ▶ Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). Hence, the name is Linear Regression ▶ Simple Linear Regression: If a single independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Simple Linear Regression. Multiple Linear regression: If more than one independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Multiple Linear Regression 	4+3	7	7
X	<p>Unsupervised machine learning</p> <ul style="list-style-type: none"> ▶ Unsupervised learning is a type of machine learning in which models are trained using unlabeled dataset and are allowed to act on that data without any supervision. ▶ The goal of unsupervised learning is to find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format. <p>Steps involved in unsupervised learning</p> <ul style="list-style-type: none"> ▶ Step 1: Collection of Necessary Data. ▶ Step 2: Training of the Model. ▶ Step 3: Model Evaluation. 	3+2+2	7	7



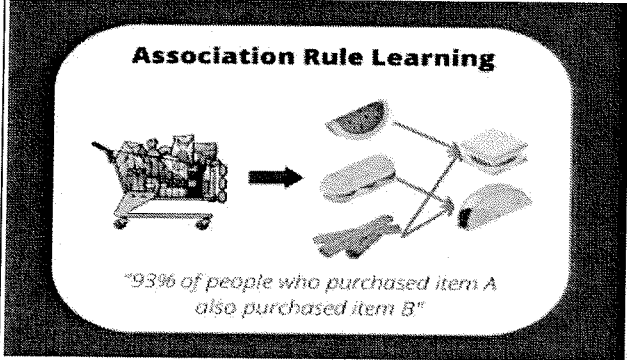
Clustering

- ▶ Clustering is a method of grouping the objects into clusters such that objects with most similarities remains into a group and has less or no similarities with the objects of another group
- ▶ Cluster analysis finds the commonalities between the data objects and categorizes them as per the presence and absence of those commonalities.



Association:

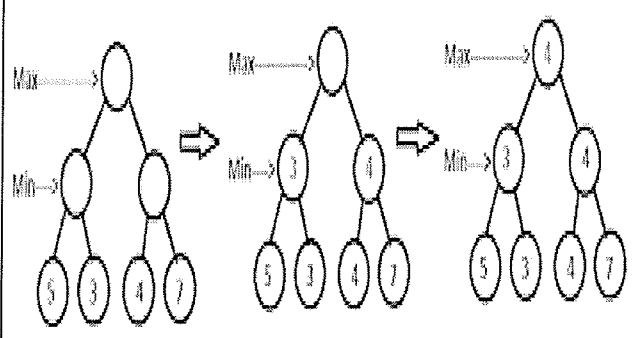
- ▶ An association rule is an unsupervised learning method which is used for finding the relationships between variables in the large database.
- ▶ It determines the set of items that occurs together in the dataset.
- ▶ Association rule makes marketing strategy more effective. Such as people who buy X item (suppose a bread) are also tend to purchase Y (Butter/Jam) item. A typical example of Association rule is Market Basket Analysis.

	 <p style="text-align: center;">Association Rule Learning</p> <p style="text-align: center;">"93% of people who purchased item A also purchased item B"</p>			
<p style="text-align: center;">XI</p>	<p>Decision tree classifier</p> <ul style="list-style-type: none"> ○ Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome. ○ In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. ○ The decisions or the test are performed on the basis of features of the given dataset. ○ <i>It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.</i> ○ It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure. ○ In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm. ○ A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees. 	<p style="text-align: center;">4+3</p>	<p style="text-align: center;">7</p>	<p style="text-align: center;">7</p>

XII	<p>Preprocessing data</p> <ul style="list-style-type: none"> ▶ Raw data is the fuel of machine learning algorithms ▶ In order to prepare the data for ingestion by machine learning algorithms, the data must be preprocessed and converted into the right format. <p>Binarization</p> <ul style="list-style-type: none"> ▶ Binarization is used to convert numerical values into Boolean values. ▶ Let's use an inbuilt method to binarize input data using 2.1 as the threshold value. ▶ As we can see here, all the values above 2.1 become 1. The remaining values become 0. <pre> 1 # Binarize data 2 data_binarized = preprocessing.Binarizer(threshold=2.1).transform(input_data) 3 print("\nBinarized data:\n", data_binarized) Binarized data: [[1. 0. 1.] [0. 1. 0.] [1. 0. 0.] [1. 0. 0.]] </pre> <p>Mean Removal Input data (Before removing mean)</p>	1+3+3	7	7

	<pre> # Print mean and standard deviation print("\nBEFORE:") print("Mean =", input_data.mean(axis=0)) print("Std deviation =", input_data.std(axis=0)) BEFORE: Mean = [3.775 -1.15 -1.3] Std deviation = [3.12039661 6.36651396 4.0620192] After mean removal # Remove mean data_scaled = preprocessing.scale(input_data) print("\nAFTER:") print("Mean =", data_scaled.mean(axis=0)) print("Std deviation =", data_scaled.std(axis=0)) AFTER: Mean = [1.11022302e-16 0.00000000e+00 2.77555756e-17] Std deviation = [1. 1. 1.] </pre>			
XIII	<p>Last coin standing game</p> <p>This is a game where we have a pile of coins and each player takes turns to take a number of coins from the pile. There is a lower and an upper bound on the number of coins that can be taken from the pile. The goal of the game is to avoid taking the last coin in the pile.</p> <pre> from easyAI import TwoPlayersGame, id_solve, Human_Player, AI_Player from easyAI.AI import TT class LastCoinStanding(TwoPlayersGame): def __init__(self, players): # Define the players. Necessary parameter. </pre>	7	7	7

<pre> self.players = players # Define who starts the game. Necessary parameter. self.nplayer = 1 # Overall number of coins in the pile self.num_coins = 25 # Define max number of coins per move self.max_coins = 4 # Define possible moves def possible_moves(self): return [str(x) for x in range(1, self.max_coins + 1)] # Remove coins def make_move(self, move): self.num_coins -= int(move) # Did the opponent take the last coin? def win(self): return self.num_coins <= 0 # Stop the game when somebody wins def is_over(self): return self.win() # Compute score def scoring(self): return 100 if self.win() else 0 # Show number of coins remaining in the pile def show(self): print(self.num_coins, 'coins left in the pile') if __name__ == "__main__": # Define the transposition table tt = TT() # Define the method LastCoinStanding.ttentry = lambda self: self.num_coins # Solve the game result, depth, move = id_solve(LastCoinStanding, range(2, 20), win_score=100, tt=tt) print(result, depth, move) # Start the game game = LastCoinStanding([AI_Player(tt), Human_Player()]) </pre>			
---	--	--	--

	game.play()			
XIV	<p style="text-align: center;">MiniMax Algorithm</p> <p>Minimax is a kind of <u>backtracking</u> algorithm that is used in decision making and game theory to find the optimal move for a player, assuming that your opponent also plays optimally. It is widely used in two player turn-based games such as Tic-Tac-Toe, Backgammon, Mancala, Chess, etc. In Minimax the two players are called maximizer and minimizer. The maximizer tries to get the highest score possible while the minimizer tries to do the opposite and get the lowest score possible.</p> <p>Every board state has a value associated with it. In a given state if the maximizer has upper hand then, the score of the board will tend to be some positive value. If the minimizer has the upper hand in that board state then it will tend to be some negative value. The values of the board are calculated by some heuristics which are unique for every type of game.</p> 	7	7	7