Multiplayer Game Connect - 4

### A Mini Project Report Submitted in partial fulfilment of the Requirement For III Semester of

**MASTER OF COMPUTER APPLICATIONS**

**Submitted By BHIMANA SREENU 20551F0009**

**Under the Esteemed Guidance of Mr. K. Praveen Kumar Assistant Professor, MCA Dept.**



**DEPARTMENT OF COMPUTER APPLICATIONS GODAVARI INSTITUTE OF ENGINEERING & TECHNOLOGY (A)**

**[Affiliated to J.N.T.U.K, Kakinada |Recognized by UGC under section 2(f) & 12(B) | NAAC ‘A+’ GRADE]**

**NH-16 CHAITANYA KNOWLEDGE CITY, RAJAHMUNDRY (A.P)**

## 2022

**GODAVARI INSTITUTE OF ENGINEERING & TECHNOLOGY (A)**

**[Affiliated to J.N.T.U.K, Kakinada |Recognized by UGC under section 2(f) & 12(B) | NAAC ‘A**+**’ GRADE]**

**NH-16 CHAITANYA KNOWLEDGE CITY, RAJAHMUNDRY (A.P)**

## 2022

**DEPARTMENT OF COMPUTER APPLICATIONS**



**CERTIFICATE**

This is to certify that this Mini project **“MULTIPLAYER GAME CONNECT - 4”** is being submitted by **BHIMANA SREENU(20551F0009)** in partial fulfilment of the requirements for III Semester of **MASTER OF COMPUTER APPLICATIONS** in the academic year 2019-2020 to the **GIET AUTONOMOUS** is a bonafide work carried out him under my guidance and supervision.

The result embodied in this project has not been submitted to any other University or Institute for the award of degree

### CO-ORDINATOR SUPERVISOR

**ACKNOWLEDGEMENT**

It is privilege for me to have under taken the project **“MULTIPLAYER GAME CONNECT - 4”**

### in GIET (A), Rajahmundry.

I avail this opportunity to express my deep sense of gratitude and heart full thanks to **Mr. K. SASI KIRAN VARMA**, Honorable managing director of **GODAVARI INSTITUTE OF ENGINEERING & TECHNOLOGY, RAJAHMUNDRY.**

I am thankful to beloved Principal **Dr. P.M.M.S.SARMA,** for permitting encouraging me in doing this project.

I am deeply indebted to **Mrs. R. TAMILKODI**, Head of the Department of Computer Applications, GIET, whose motivation in the field of software development made me overcome all the hardships during the course of study.

I am heartily thankful to my internal guide **Mr. K. Praveen Kumar** Assistant Professor, MCA dept, GIET, for valuable guidance, keen interest, open minded discussion and his constant encouragement throughout the project work.

Finally I would like to extend my heartfelt thanks to my **TEACHING AND NON TEACHING STAFF** who had shared their technical knowledge and their encouragement were always there as a source of strength and inspiration.

Although a leaflet title “Acknowledgement” cannot represent my true feelings for all these persons, I feel very much thankful to all of them and also to my **PARENTS** and **FRIENDS** for encouraging and giving me all the moral support required for making this endeavour a reality.

**BHIMANA SREENU**

### (20551F0009)

**DECLARATION**

I **BHIMANA SREENU (PIN 20551F0009)** declare that the mini project title **“MULTIPLAYER GAME CONNECT - 4 ”** is a bonafide work carried out by me and has not been submitted to any other university or college for the award of degree.

### BHIMANA SREENU (20551F0009)

**TABLE OF CONTENTS**

**Title Page.No**

**CHAPTER-1: INTRODUCTION**

* 1. Purpose1 1-2
  2. [Scope 2](#_TOC_250013)

[CHAPTER-2: SYSTEMANALYSIS](#_TOC_250012)

* 1. [Existing System 3](#_TOC_250011)
  2. [Proposed System 3-7](#_TOC_250010)

[CHAPTER-3: SYSTEM REQUIREMENTS](#_TOC_250009)

* 1. [Functional Requirements 8](#_TOC_250008)
  2. Non-Functional Requirements 8
  3. [Software Requirements 9](#_TOC_250007)
  4. [Hardware Requirements 9](#_TOC_250006)

CHAPTER-4: SYSTEM ADESIGN 10-12

[CHAPTER-5: SYSTEM IMPLEMENTATION](#_TOC_250005)

* 1. [Technology Description 13-14](#_TOC_250004)
  2. [Sample Code 14-20](#_TOC_250003)

[CHAPTER-6: SYSTEM TESTING 21-22](#_TOC_250002)

[CHAPTER-7: RESULTS 23-27](#_TOC_250001)

CHAPTER-8: CONCLUSION & FUTURESCOPE 28

[References 29](#_TOC_250000)

**ABSTRACT**

Connect Four is a two players game which takes place on a 7\*6 rectangular board placed vertically between them. One player has 21 yellow coins and the other 21 red coins. Each player can drop a coin at the top of the board in one of the seven columns ; the coin falls down and fills the lower unoccupied square. Of course a player cannot drop a coin in a certain column if it’s already full (i.e., it already contains six coins).

Even if there’s no rule about who begins first, we assume, as in chess that the lighter side makes the first move. We also use the chess notation to represent a square on the board. That is, we number rows from 1 to 7 starting from the bottom and the columns from A to G starting from the leftmost.

The object of the game is to connect four coins vertically, horizontally or diagonally. If the board is filled and no one has aligned four coins then the game is drawn (i.e., after 42 moves if no one wins).

# CHAPTER-1 INTRODUCTION

## Purpose

Artificial Intelligent (AI) is the computational study of intelligent thought and behavior. According to Russell and Norvig (2010) definition of artificial intelligence are classified into four different categories that are whether they are concerned with thought processes or with behavior and whether they measure success against human performance or against "rationality", an ideal standard of behavior . Also, Intelligent is best defined as the ability to learn or understand from experience; the ability to acquire and retain knowledge; mental ability; use of the faculty of reason in solving problems, directing conduct etc. efficiently in this system, user explore the game playing of connect four using artificial intelligent. In connect four, two players compete against each other until one players successfully connects four tokens of same color in a row either horizontally, vertically, or diagonally. My goal is to create a competent artificial intelligence that plays against a human opponent. User used python language to implement the game playing.

User have use alpha beta pruning that uses a heuristic in a static evaluation function (SEF). The static evaluation function is used to find the best move in game playing by measuring a board configuration and how good a potential move is by using weights. Similarly, minimax is used to search a game tree to a depth of five. Alpha beta pruning eliminates calculation of some static evaluation and unnecessary traverse. There are different sections in this articles that will go in detail on each major topic such as the game, algorithms etc.

## Scope

Connect four is a two-player game played on a board consisting of seven columns, each six- cell high. The goal of the game for a player is to connect four of their tokens in a row either horizontally, vertically, or diagonally. Tokens are usually red and blue colors. To place a token in a board players drop it from the top for the particular column. The first open space available in that column will be filled by the token.Initially, game begin with empty board. Each player owns a certain number of tokens of same color. Each player takes a turn to drop the token into the board. The first player to get four tokens of same color in a row wins the game. Players take alternate turns to put the tokens into the board. If none of the players successfully place four tokens of same color in a row, game ends in a draw. A timer can be used to give a player only certain time to think and make a move.

# CHAPTER-2 SYSTEM ANALYSIS

## Existing System

Existing system consists less features compare to the proposed system and one or two players sitting at a single computer, which was being used only to play the game concept of time sharing. It consists complex algorithm.

## Disadvantages

* + - Less Features
    - Complex algorithms

## Proposed System

Program deals with artificial intelligence. The computer is required to use it's intelligence against The each move by the user which will be dependent on the developed algorithm. The more the complexity and efficiency of the algorithm the better will be tougher for the user to prove him/her self smarter than the machine.

The game is categorized as a zero-sum game. Therefore, the minimax algorithm, which is a decision rule used in AI, can be applied. The project goal is to investigate how a decision tree is applied using the minimax algorithm in this game by Artificial Intelligence.

## Advantages

People approached user AI design with a focus on how a human player intuitively analyzes game states and makes game play decision. Although there are existing algorithms capable of always making the most optimal move, people chose not to go in that direction due to the excessive usage of memory and recursive functions for the sake of traversing an exponentially increasing number of game states, which is in practice very difficult to implement in hardware, instead people opted for an iterative design approach , starting with the most basic defensive and offensive behaviors based on a custom heuristic people designed which closely follows human intuition when playing the game. This allowed people to create a reasonably challenge

AI that implements move anticipation using purely combinational logic such that the move determination could be completed within a single clock cycle. This is also described in greater detail within the “ AI Design Iterations” section.

In this era of artificial experiences , connect 4 games prove very useful in helping today’s kids learn valuable life skills like strategy, visual perception , and problem solving . Let’s learn how kids can use them to learn on smart devices.

## Modules Description

### Existing System modules

* + - **players**(tuple of player) – The two players. The first one is first to play.
    - **n rows** – The number of rows of the board. Default is 6.
    - **n cols (int)** – The number of columns of the board. Default is 7.
    - **to win(int)** – The number of aligned pieces required to win the game. Default is 4.

### Player1

* + - Player
    - The first player.

### Player2

* + - Player
    - The second player.

### board

* + - Board
    - The board

### To win

* + - int
    - The number of aligned pieces required to win the game.

### Check winner()

* + - Check if there’s a winner at the current game state.
    - Returns: The winner. If no player has won yet, none is returned.
    - Return type: player

### run()

* + - Run a game session between the two players.
    - Returns: The winner
    - Return type: player

### Proposed system modules

**play(board)**

Choose a column to play on based on the minmax algorithm.

**Parameters: board**(Board)- The current board.

**Returns:** The column to play on.

**Return type:** (int)

### Utility(board)

The utility function to evaluate the *goodness* of a board for the player.

Right now, the evaluation is as follows:

* If the player wins, ∞∞ is returned.
* If he loses, −∞−∞ is returned.
* If there is no winner, the utility value starts from 00 and incremented by l2l2 for each sequence of length ll that could lead to a win for the player, and decremented by the same amount for each sequence that could lead to a win for the opponent.

|  |  |
| --- | --- |
| **Parameters:** | **board** (Board) – The board to evaluate. |
| **Returns:** | **The estimated utility of the board.** |

|  |  |
| --- | --- |
|  |  |

**FEASIBILITY STUDY**

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time.

There are aspects in the feasibility study portion of the preliminary investigation:

* Technical Feasibility
* Operational Feasibility
* Economical Feasibility

## ECONOMIC FEASIBILITY

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC, There is nominal expenditure and economical feasibility for certain.

## OPERATIONAL FEASIBILITY

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization’s operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

* Is there sufficient support for the management from the users?
* Will the system be used and work properly if it is being developed and implemented?
* Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration.

So there is no question of resistance from the users that can undermine the possible application benefits.

The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

## TECHNICAL FEASIBILITY

The technical issue usually raised during the feasibility stage of the investigation includes the following:

* Does the necessary technology exist to do what is suggested?
* Do the proposed equipments have the technical capacity to hold the data required to use the new system?
* Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
* Can the system be upgraded if developed?
* Are there technical guarantees of accuracy, reliability, ease of access and data security? Earlier no system existed to cater to the needs of ‘Secure Infrastructure Implementation

System’. The current system developed is technically feasible. It is a web based user interface for audit workflow at NIC-CSD. Thus it provides an easy access to the users. The database’s purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hard requirements for the development of this project are not many and are already available inhouse at NIC or are available as free as open source.

# CHAPTER-3 SYSTEM REQUIREMENTS

## 3.1 FUNCTIONAL REQUIREMENTS

### Computer

Write a function suggestMove1(board,user) which takes as inputs a list board representing the Connect Four board and an integer who with possible values 1 or 2. The function returns an integer between 0,1,...,6 corresponding to a column index of the board into which player number should insert their "disc". This column index is determined as follows: First check if among all valid moves of player number who there is a move which leads to an immediate win of this player. In this case, return such a winning move. If there is no winning move for player number who, we will try to prevent the other player from winning. This is done by checking if there is a winning move for the other player and returning it Otherwise, if there is no immediate winning move for both players, the function simply returns a valid move (for example, the first one in the list of valid moves).

### Player1

The pieces fall straight down, occupying the lowest available space within the column. The objective of the game is to be the first to form a horizontal, vertical, or diagonal line of four of one's own tokens. Connect Four is a solved game. The first player can always win by playing the right moves.

### Player2

Write a function new Game(player1, player2) which takes two string parameters player1 and player2 corresponding to the two players' names. The function returns a game dictionary as specified in Task 0. In this dictionary all the positions of the board are empty (i.e., all are set to the integer 0), the players' names are set to the input parameters of the function, and the variable who is set to the integer 1 (in a new game, player 1 will always make the first move).

## Software Requirements

Language : Python

Technology : AI(Artificial Intelligence)

Frontend : Python3

Backend : Python flask, SQL

IDE : html Sublime text

Operating System : windows

## Hardware Requirements

Processor : i3

Hard Disk : 80GB

RAM : 8GB

# CHAPTER-4 SYSTEM DESIGN

The purpose of the design phase is to plan a solution of the problem specified by the requirement document. This phase is the first step in moving from the problem domain to the solution domain. In other words, starting with what is needed, design takes us toward how to satisfy the needs. The design of a system is perhaps the most critical factor affection the quality of the software; it has a major impact on the later phase, particularly testing, maintenance. The output of this phase is the design document. This document is similar to a blueprint for the solution and is used later during implementation, testing and maintenance. The design activity is often divided into two separate phases System Design and Detailed Design. System Design also called top-level design aims to identify the modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired results.

## System Model

### Introduction to UML

The unified Modeling Language (UML) is a standard language for writing software blueprints. The UML may be used to visualize, specify , construct and document the artifacts of software-intensive system.

The goal of UML is to provide a standard notation that can be used by all object - oriented methods and to select and integrate the best elements .UML is itself does not prescribe or advice on how to use that notation in a software development process or as part of an object - design methodology. The UML is more than just bunch of graphical symbols. Rather , behind each symbol in the UML notation is well-defined semantics.

The system development focuses on three different models of the system.

* Functional model
* Object model
* Dynamic model

**Functional model** in UML is represented with use case diagrams , describing the functionality of the system from user point of view.

**Object model** in UML is represented with class diagrams , describing the structure of the system in terms of objects , attributes , associations and operations

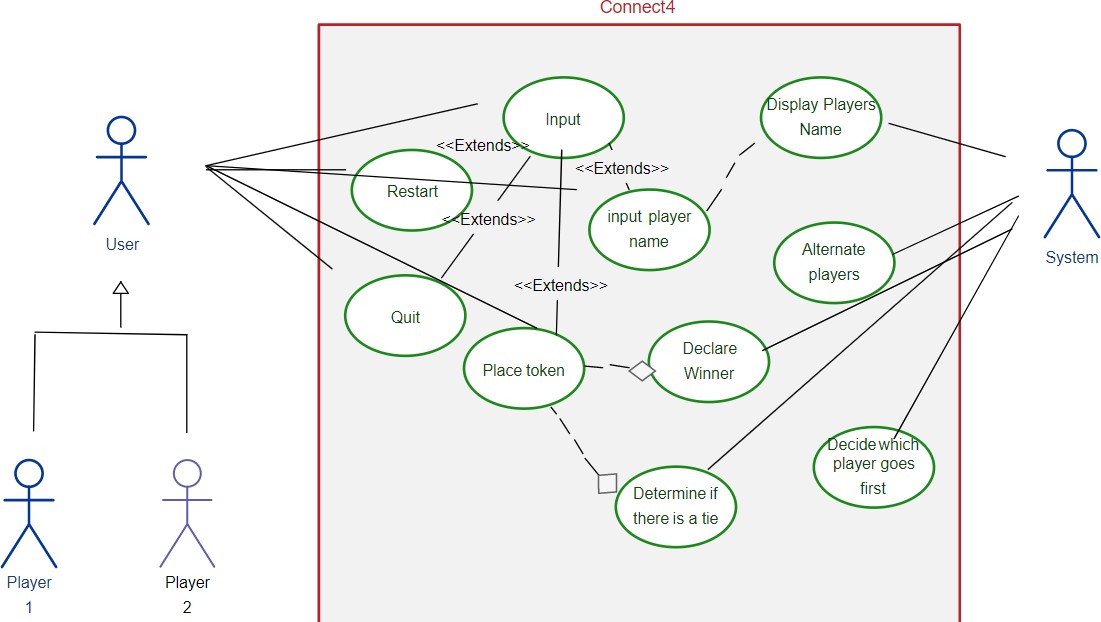
## Data Flow Diagrams

A graphical tool used to describe and analyze the moment of data through a system manual or automated including the process, stores of data, and delays in the system. Data Flow Diagrams are the central tool and the basis from which other components are developed. The transformation of data from input to output, through processes, may be described logically and independently of the physical components associated with the system. The DFD is also know as a data flow graph or a bubble chart.

DFDs are the model of the proposed system. They clearly should show the requirements on which the new system should be built. Later during design activity this is taken as the basis for drawing the system’s structure charts. The Basic Notation used to create a DFD’s are as follows:

1. **Dataflow:** Data move in a specific direction from an origin to a destination.
2. **Process:** People, procedures, or devices that use or produce (Transform) Data. The physical component is not identified.

**Class diagram**



# CHAPTER-5 SYSTEM IMPLEMENTATION

## Technology Description

**Python:**

Python is an interpreted high-level general–purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, Including structured(particularly procedural),object-oriented and functional programming. It is often described as a “batteries included” language due to its comprehensive standard library.

Guido van Rossum began working on Python in the late 1960’s as a successor to the ABC programming language, and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features, such as list comprehensions and a cycle-detecting garbage collection system (in addition to reference counting).Python 3.0 was released in 2008 and was a major revision of the language that is not completely backward-compatible. Python 2 was discontinue with version 2.7.18 in 2020 .



res of Python:

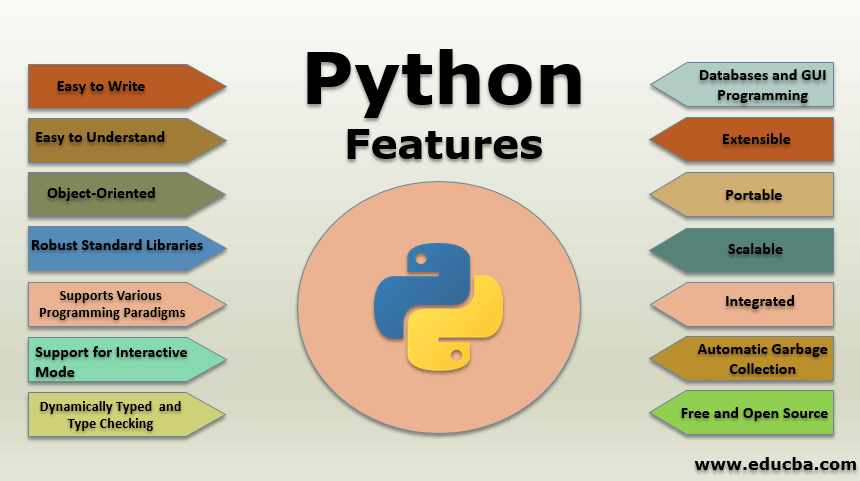
Python is a dynamic, high level, free open source and interpreted programming language. It supports object-oriented programming as well as procedural oriented programming as well as procedural oriented programming.

In Python, we don’t need to declare the type of variable because it is a dynamically typed language.

For example,x=10

Here, x can be anything such as String,int etc.,

There are many features in Python,some of which are discussed below-



## Easy to Write

* + It is easy to write as compared to other programming languages.It’s syntax is straightforward and much the same as the English language.
  + There is no use of the semicolon or curly-bracet,the indentation defines the code block.

## Object-Oriented Language

* + Python supports object-oriented language and concepts of classes and objects come into existence.
  + It supports inheritance, polymorphism, and encapsulation, etc.,

## Robust standard Library

* + It provides a vast range of libraries for the various fields such as machine learning, web developer, and also for scripting.
  + There are various machine learning libraries, such as Tensor flow, Pandas, Numpy, Keras, and Pytorch.

## Databases and GUI Programming

* + Graphical User Interface is used for the developing Desktop applications, PyQT5, Tkinter, Kivy are libraries which are used for developing the web application.

## Extensible

* + It implies that other languages such as C/C++ can be used to compile the code and thus it can be used further in our Python code.

## Portable

* + Python language is also a portable language. For example, if we have python code for windows and if we want to run this code on other platforms such as Linux, Unix, and Mac then we do not need to change it, we can run this code on any platform.

## Integrated

* + Python is also an integrated language because we can easily integrated python with other languages like C,C++ etc.,

## Free and Open Source

* + Python language is freely available at the official website and you can download it from the given link below click on the Download python keyword.
  + Since it is open source, this means that source code is also available to the public. So you can download it as ,use it as well as share it.

## Applications Of Python

Python is known for its general-purpose nature that makes it applicable in almost every domain of software development. Python makes its presence in every emerging field. It is the fastest- growing programming language and can develop any application.

Here, we are specifying application areas where Python can be applied.

1. Web Applications
2. Desktop GUI Applications
3. Console Based Applications
4. Software Development
5. Scientific and Numeric
6. Business Applications
7. Audio or Video-Based Applications
8. 3D CAD Applications
9. Enterprise Applications
10. Image Processing Applications
11. Embedded Applications
12. Machine Learning and Artificial Intelligence
13. Game Development.

**Libraries Of Python** 1.Numpy module 2.Pygame module 3.Python sys module 4.Python math module **1.Numpy module**

NumPy stands for Numerical Python. NumPy is a Python library used for working with

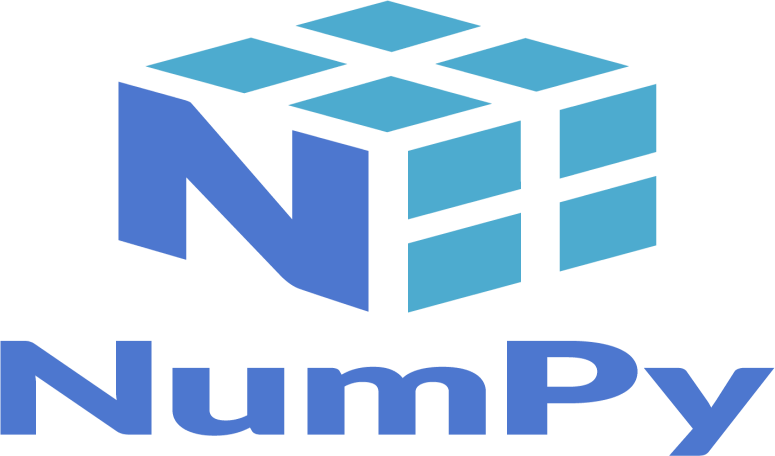
arrays.It also has fuctions for working in domain of linear algebra, fourier transform,and matrices.It is an open source project you can use it freely. NumPy is a python library that provides a simple yet powerful data structure the n-dimensional array.

If you don’t have NumPy already pre-installed on your system, type the following command in your windows cmd:

C:\users\Admin>pip install numpy

When you call the statement import numpy as np,you are shortening the phrase “NumPy” to “np” to make your code easier to read.It also helps to avoid namespace issues.

Import numpy as as np



## Pygame module

Pygame is a free and open-source cross-platform library for the development of multimedia applications like video games using Python.

It uses the Simple Direct Media Layer library and several other popular libraries to abstract the most common functions, making writing these programs a more intuitive task.

If you don’t have Pygame already pre-installed on your system, type the following command in your window’s cmd:

C:\users\Admin>pip install numpy



## Python Sys module

The Python Sys module provides functions and variables which are used to manipulate different parts of Python Runtime Environment. It lets us access system-specific parameters and functions.import sys. First, we have to import the sys module in our program before running any functions sys.modules.



## Python math module

Some of the most popular mathematical functions are defined in the math module. These include trigonometric functions, logarithmic functions, angle conversion functions etc., In addition ,two mathematical constants are also defined in this module. If you don’t have math already pre- installed on your system, type the following in your window’s cmd:

C:\users\Admin>pip install maths

## Artificial Intelligence

Artificial intelligence (AI) broadly refers to any human- like behavior displayed by a machine or system. In AI’s most basic form, computers are programmed to “mimic’ huma behavior using extensive data from past examples of similar behavior. This can range from recognizing differences between a cat and a bird to performing complex activities.

**Python Flask**

So there was a requirement for making a quick backend API using an tech stack of your choice for the game Connect 4.The aim **for** both players is to make a straight line of **four** own pieces; the line can be vertical, horizontal or diagonal.

Before starting, players decide randomly which of them will be the beginner; moves are made alternatively, one by turn. In the API we have to test if the call made by the user is valid or invalid and if the user is a winner we need to update the same in the API call giving the appropriate response.

First of all we need to install Flask for our usage. Install Flask using the following command. After installing open a directory/folder and start developing.pip install Flask

After creating and navigating to the folder we just created we need to create a file that will be connecting with the app.py or main.py later to provide the functionalities needed for this game to run properly. So create a file named **game.py** and copy the contents from the code provided below

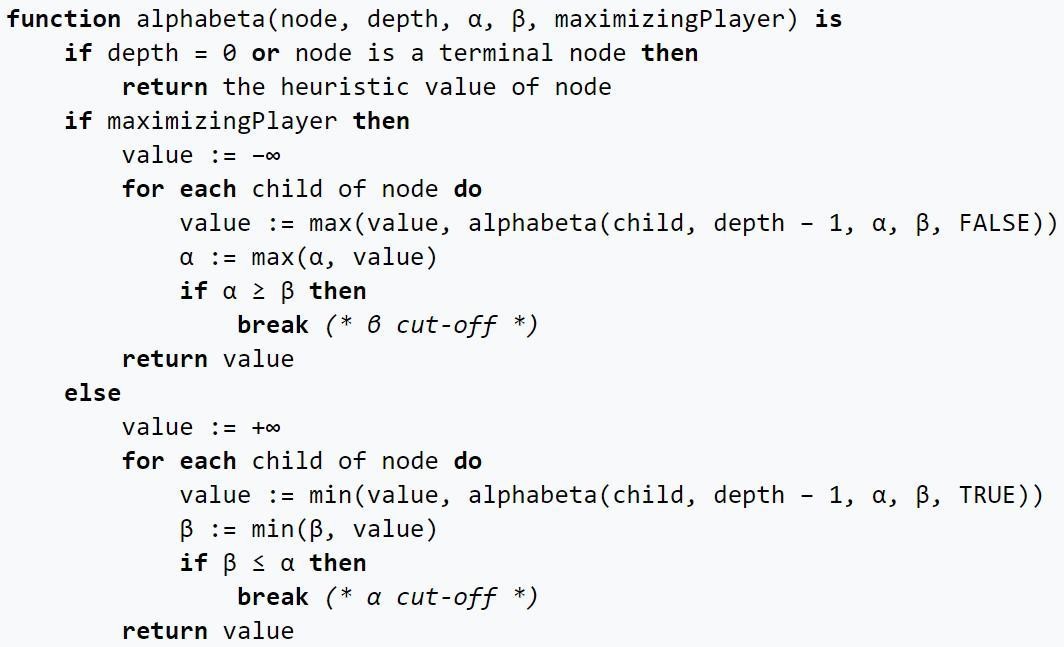
## SQL

SQL is the standard language for dealing with Relational Databases. SQL can be used to insert, search, update, and delete records. SQL can do lots of other operations , including optimizing and maintenance of databases.

## SAMPLE CODE

### AI in Connect Four — Implementing Minimax

Below is a python snippet of Minimax algorithm implementation in Connect Four. In the code, we extend the original Minimax algorithm by adding the Alpha-beta pruning strategy to improve the computational speed and save memory. The figure below is a pseudocode for the alpha-beta minimax algorithm.



|  |
| --- |
| def minimax(board, depth, alpha, beta, maximizingPlayer): |
| valid\_locations = get\_valid\_locations(board) |

|  |
| --- |
| is\_terminal = is\_terminal\_node(board) |
| if depth == 0 or is\_terminal: |
| if is\_terminal: |
| if winning\_move(board, AI\_PIECE): |
| return (None, 100000000000000) |
| elif winning\_move(board, PLAYER\_PIECE): |
| return (None, -10000000000000) |
| else: # Game is over, no more valid moves |
| return (None, 0) |
| else: # Depth is zero |
| return (None, score\_position(board, AI\_PIECE)) |
| if maximizingPlayer: |
| value = -math.inf |
| column = random.choice(valid\_locations) |
| for col in valid\_locations: |
| row = get\_next\_open\_row(board, col) |
| b\_copy = board.copy() |
| drop\_piece(b\_copy, row, col, AI\_PIECE) |
| new\_score = minimax(b\_copy, depth-1, alpha,  beta, False) |
| if new\_score > value: |

|  |
| --- |
| value = new\_score |

|  |
| --- |
| alpha = max(alpha, value) |
| if alpha >= beta: |
| break |
| return column, value |
| else:#Minimizingplayer |
| value = math.inf |
| column = random.choice(valid\_locations) |
| for col in valid\_locations: |
| row = get\_next\_open\_row(board, col) |
| b\_copy = board.copy() |
| drop\_piece(b\_copy, row, col, PLAYER\_PIECE) |
| new\_score = minimax(b\_copy, depth-1, alpha,  beta, True) |
| if new\_score < value: |
| value = new\_score |
| column = col |
| beta = min(beta, value) |
| if alpha >= beta: |
| break |
| return column, value |

Let us take the **maximizing Player** from the code above as an example (From line 136 to line 150). First, the program will look at all valid locations from each column, recursively getting the new score calculated in the look-up table (will be explained later), and finally update the optimal value from the child nodes. Notice that the alpha here in this section is the new\_score, and when it is greater than the current value, it will stop performing the recursion and update the new value.

32

As mentioned above, the look-up table is calculated according to the evaluate\_window function below. Here, the window size is set to four since we are looking for connections of four discs. Considering a reward and punishment scheme in this game. If four discs are connected, it is rewarded for a high positive score (100 in this case). When three pieces are connected, it has a score less than the case when four discs are connected. When two pieces are connected, it gets a lower score than the case of three discs connected. Finally, when the opponent has three pieces connected, the player will get a punishment by receiving a negative score. Indicating that it is not an optimal move for the current player.

With the scoring criteria set, the program now needs to calculate all scores for each possible move for each player during the play. The function score\_position performs this part from the below code snippet. The AI player will then take advantage of this function to predict an optimal move.

|  |
| --- |
| def score\_position(board, piece): |
| score = 0 |
| ## Score centercolumn |
| center\_array = [int(i) for i in list(board[:, COLUMN\_COUNT//2])] |
| center\_count =  center\_array.count(piece) |
| score += center\_count \* 3 |
| ## Score Horizontal |
| for r in range(ROW\_COUNT): |

|  |
| --- |
| row\_array = [int(i) for i in  list(board[r,:])] |
| For c in range(COLUMN\_COUNT-3): |
| window =  row\_array[c:c+WINDOW\_LENGTH] |
| score +=  evaluate\_window(window, piece) |
| ## Score Vertical |
| for c in range(COLUMN\_COUNT): |
| col\_array = [int(i) for i in list(board[:,c])] |
| for r in range(ROW\_COUNT-  3): |
| window =  col\_array[r:r+WINDOW\_LENGTH] |
| score +=  evaluate\_window(window, piece) |
| ## Score posiive sloped diagonal |
| for r in range(ROW\_COUNT-3): |
| for c in range(COLUMN\_COUNT-3): |

|  |
| --- |
| window =  [board[r+i][c+i] for i in range(WINDOW\_LENGTH)] |
| score +=  evaluate\_window(window, piece) |
| for r in range(ROW\_COUNT-3): |
| for c in range(COLUMN\_COUNT-3): |
| window = [board[r+3-  i][c+i] for i in range(WINDOW\_LENGTH)] |
| score +=  evaluate\_window(window, piece) |
| return score |

# CHAPTER-6 SYSTEM TESTING

## TESTING METHODOLOGIES

The following are the Testing Methodologies:

### Unit Testing.

* **Integration Testing.**
* **User Acceptance Testing.**
* **Output Testing.**
* **Validation Testing.**

**Unit Testing**

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module’s control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

## Integration Testing

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

### The following are the types of Integration Testing:

**1)Top Down Integration**

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

### Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

* + - The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
    - A driver (i.e.) the control program for testing is written to coordinate test case input and output.
    - The cluster is tested.
    - Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

### User Acceptance Testing

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

### Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways

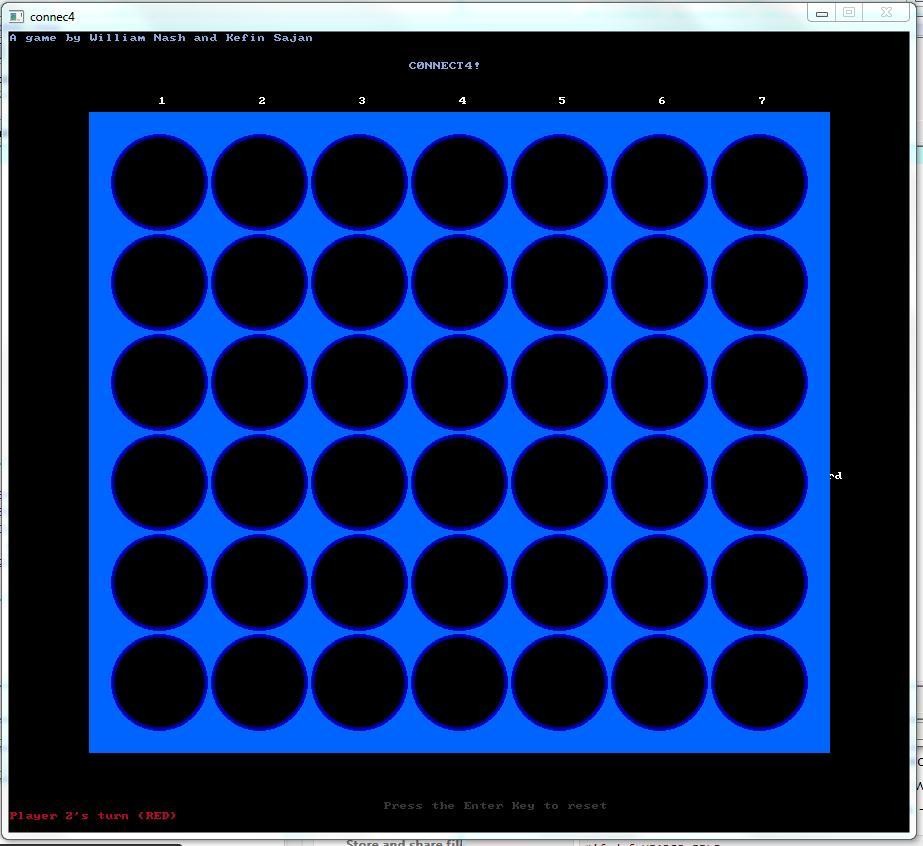
– one is on screen and another in printed format.

# CHAPTER-7 RESULTS

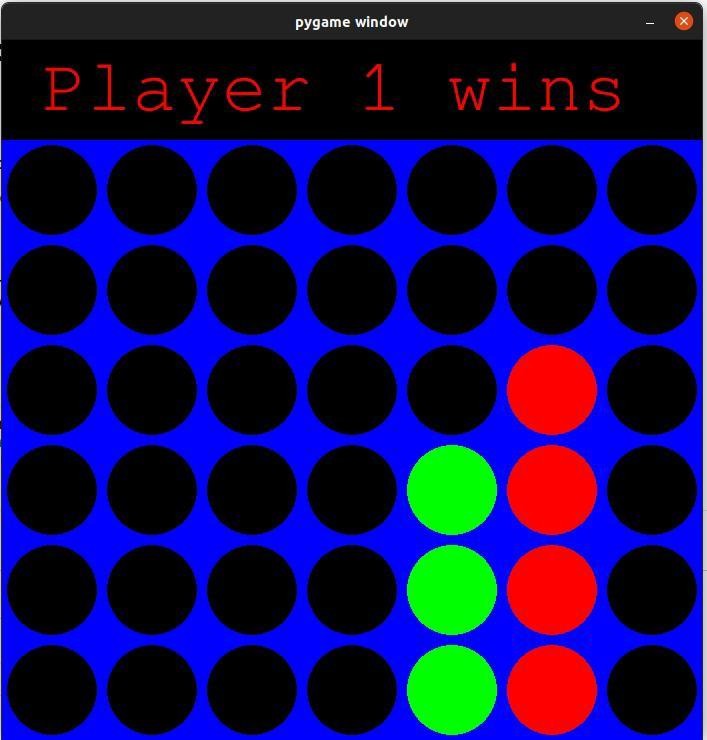
## Four Game Connect



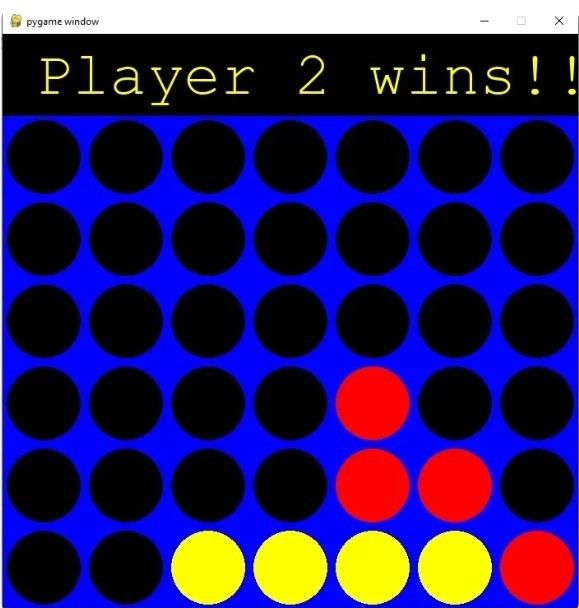
* 1. **Layout of Connect 4**



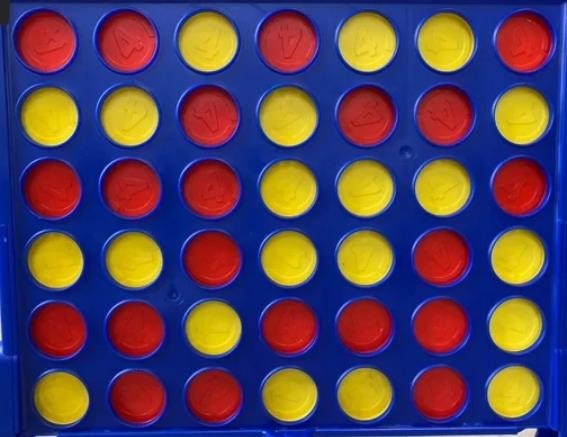
* 1. **Player one wins in connect 4**



* 1. **Player two wins in connect4**



* 1. **Connect 4 tie result**



**CHAPTER-8**

**CONCLUSION & FUTURE SCOPE**

Looking at how many times AI has beaten human players in this game, I realized that it wins by rationality and loads of information. In this project, the AI player uses a minimax algorithm to check for optimal moves in advance to outperform human players by knowing all possible moves rationally. Interestingly, when tuning the number of depths at the minimax function from high (6 for example) to low (2 for example), the AI player may perform worse. Nevertheless, the strategy and algorithm applied in this project have been proved to be working and performing amazing results.

The minimax algorithm with alpha-beta pruning is an excellent choice of algorithm for the game of complete information, zero sum games, two player games such as connect four. Because the depth of the tree is five it is easy for computers to determine the moves ahead and play. As explain above, connect four is a very human-like artificially intelligent program. It is very difficult for beginners to easily beat computer while playing this program. With more time permitted, I can improve the minimax alpha beta pruning to make this game more competitive to play against an expert.

# REFERENCES

1. Galli. (n.d.). KeithGalli/Connect4-Python. GitHub. [https://github.com/KeithGalli/Connect4- Python](https://github.com/KeithGalli/Connect4-Python) Nasa, R., Didwania, R., Maji, S., & Kumar, V. (2018). Alpha-beta pruning in mini-max algorithm–an optimized approach for a connect-4 game. Int. Res. J. Eng. Technol, 1637–1641.
2. Ginsberg, M. Essential of Artificial Intelligence. Morgan Kaufmann Publishers, San Francisco, 1993.
3. Lily Newman - [https://gizmodo.com/heres-how-to-win every-time-at-connect-four- 1474572099](https://gizmodo.com/heres-how-to-win%20every-time-at-connect-four-1474572099)
4. Milton Bradley, 1974. Connect Four (also known as Plot Four, Find Four, Four in a Row and Four in a Wikipedia; [http://en.wikipedia.org/wiki/Connect\_Fourline),](http://en.wikipedia.org/wiki/Connect_Fourline))
5. Wikipedia, 2009. Waterfall Mode[l. http://en.wikipedia.org/wiki/Waterfall](http://en.wikipedia.org/wiki/Waterfall) model
6. Adams, R., E. Ibsen and C. Zhang, 2003. A Connect Four Playing AI Agent: Algorithm and Creation Process. <http://www.ccs.neu.edu/home/eclip5e/classes/csu5> 20/SmartConnectFour-

Final Paper-v1.0.doc

1. Pomakis,K.,2005.Connect-4

Algorithm; <http://www.pomakis.com/c4/connect_generic/c4.txt> 5. Labib, B., 2003. Connect4 using Alpha-Beta

Search algo[rithm. http://www.codeproject.com/KB/mobile/Connect4](http://www.codeproject.com/KB/mobile/Connect4) AB.aspx

1. Padhy, N.P., 2005. State Space Search: Implementation and Applications. In: Artificial Intelligence and Intelligent Systems. Oxford University Press, New York, pp: 161-171.
2. Russell, S.J. and P. Norvig, 1995. Informed Search Methods. In: Artificial Intelligence: A Modern Approach. Upper Saddle River, Prentice-Hall, New Jersey, pp: 96-98.
3. Matthews, J., 2000. A\* for the Masses. <http://www.generation5.org/content/2000/astar.asp>