

# Bahria University,

## Karachi Campus



### LAB EXPERIMENT NO.

02

### LIST OF TASKS

TASK NO	OBJECTIVE
1.	<b>Implement Tic Tac Toe game by using Min Max Algorithm (Adversial Search) which suggest user a best move</b>
2.	<b>Implement a text summarization model using Transformers Scenario:</b> As a natural language processing (NLP) researcher, Your task is to utilize the Transformers library in Python to build and train a summarization model. The model should be able to take a long text document as input and generate a concise summary that captures the key information and main ideas.
3.	<b>Convert images to sketches using OpenCV Scenario:</b> Your task is to create a Python script that uses the OpenCV library to convert regular images into sketches. The script should allow users to select an image file, apply appropriate filters and transformations to convert it into a sketch-like image, and save the resulting image to disk.
4.	<b>Build a web scraper using BeautifulSoup Scenario:</b> Your task is to develop a Python script that uses the BeautifulSoup library to scrape product information from competitor websites. The script should be able to extract data such as product names, descriptions, prices, and images from the target websites and store the data in a structured format (e.g., CSV or JSON) for further analysis.
5.	<b>Automate WhatsApp messaging using PyWhatKit Scenario:</b> Your task is to create a Python script that uses the PyWhatKit library to automate the sending of messages and images through WhatsApp. The script should allow users to schedule the sending of messages or images to one or more contacts at specific times or intervals.
6.	<b>Develop a text-to-speech application using pyttsx3 Scenario:</b> Your task is to create a Python application that uses the pyttsx3 library to convert text into spoken words. The application should allow users to input text, select voice settings (e.g., language, gender, rate), and generate audio output that can be played or saved to a file.

**Submitted On:**

**4/5/2024**

**TASK NO 1: Implement Tic Tac Toe game by using Min Max Algorithm (Adversial Search) which suggest user a best move.**

```

def ConstBoard(board):
    print("Current State Of Board : \n\n");
    for i in range (0,9):
        if((i>0) and (i%3)==0):
            print("\n");
        if(board[i]==0):
            print("- ",end=" ");
        if (board[i]==1):
            print("O ",end=" ");
        if(board[i]==-1):
            print("X ",end=" ");
    print("");
def User1Turn(board):
    pos=input("Enter X's position from [1...9]: ");
    pos=int(pos);
    if(board[pos-1]!=0):
        print("Wrong Move!!!");
        exit(0);
    board[pos-1]=-1;
def User2Turn(board):
    pos=input("Enter O's position from [1...9]: ");
    pos=int(pos);
    if(board[pos-1]!=0):
        print("Wrong Move!!!");
        exit(0);
    board[pos-1]=1;
#MinMax function.
def minimax(board,player):
    x=analyzeboard(board);
    if(x!=0):
        return (x*player);
    pos=-1;
    value=-2;
    for i in range(0,9):
        if(board[i]==0):
            board[i]=player;
            score=-minimax(board,(player*-1));
            if(score>value):
                value=score;
                pos=i;
            board[i]=0;
    if(pos==-1):
        return 0;
    return value;
def CompTurn(board):
    pos=-1;
    value=-2;
    for i in range(0,9):
        if(board[i]==0):
            board[i]=1;
            score=-minimax(board, -1);
            if(score>value):
                value=score;
                pos=i;
            board[pos]=1;
def analyzeboard(board):
    cb=[[0,1,2],[3,4,5],[6,7,8],[0,3,6],[1,4,7],[2,5,8],[0,4,8],[2,4,6]];
    for i in range(0,8):
        if(board[cb[i][0]] !=0 and board[cb[i][1]] == board[cb[i][2]] and board[cb[i][0]] == board[cb[i][1]] and board[cb[i][0]] == board[cb[i][2]]):
            return board[cb[i][2]];
    return 0;
def main():
    board=[0,0,0,0,0,0,0,0,0];
    print("Computer : O Vs. You : X");
    player= input("Enter to play 1(st) or 2(nd) :");
    player = int(player);
    for i in range (0,9):
        if(analyzeboard(board)!=0):
            break;
        if((i+player)%2==0):
            CompTurn(board);
        else:
            ConstBoard(board);
            User1Turn(board);
    x=analyzeboard(board);
    if(x==0):
        ConstBoard(board);
        print("Draw!!!")
    if(x==-1):
        ConstBoard(board);
        print("YOU Wins!!! MINMAX Loose !!!")
    if(x==1):
        ConstBoard(board);
        print("MINMAX Loose!!! YOU Wins !!!")
    main()

```

**Output:**

```

Computer : O Vs. You : X
Enter to play 1(st) or 2(nd) :1
Current State Of Board :

- - -
- - -
- - -
Enter X's position from [1...9]: 3
Current State Of Board :

- - X
- O -
- - -
Enter X's position from [1...9]: 4
Current State Of Board :

O - X
X O -
- - -
Enter X's position from [1...9]: 9
Current State Of Board :

O - X
X O O
- - X
Enter X's position from [1...9]: 2
Current State Of Board :

O X X
X O O
O - X
Enter X's position from [1...9]: 8
Current State Of Board :

O X X
X O O
O X X
Draw!!!

```