**CSE 537 Project 2 Game Search**

**Group Details:**

**Shrey Shah 109247887**

**Rutuja Sudam Marathe 109967750**

**Note: For running all the codes we have defined all the run\_games() in line 256 of lab3.py**

**Description:**

1. Minimax Algorithm:

Function minimax() in basicplayer.py calls function recursiveMinimax() that implements minimax algorithm. Inputs of this function are board, depth , get next move function, terminal function and variable that tells whether its max or min step.

1. Alphabeta Algorithm:

Function alpha\_beta\_search() in lab3.py calls function recursiveAlphaBeta() that implements alphabeta algorithm. Inputs of this function are board, depth , get next move function, terminal function and variable that tells whether its max or min step.

1. Minimax algorithm and Alphabeta algorithm with varying depth:

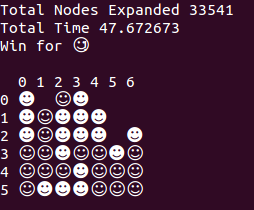
We are passing depth as parameter of run\_game function. We have variable k in object ConnectFourBoard that stores depth. Default value is 4.

1. Longest Streak Problem:

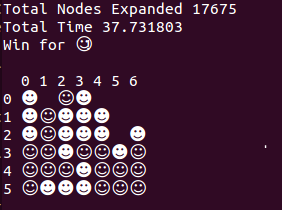
We have defined new run\_game\_longest\_streak() in connectFour.py and a new longest\_streak\_player which takes care of all the terminal condition's , is\_game\_over, and is\_tie condition's.

**Results:**

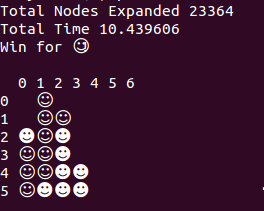
1. Minimax Algorithm with depth = 4



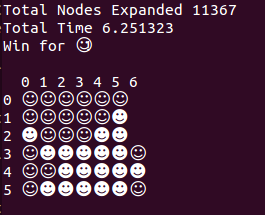
1. Alpha Beta Algorithm with depth = 4



1. Mini Max Algorithm with depth = 5



1. Alpha beta Algorithm with depth = 6



1. Longest Streak Problem:

