**Main Flowchart**

Checkmate = False

INITIATE Pygame Window

BLIT Board ON Pygame Window

BLIT Save\_Button ON Pygame Window

BLIT Load\_Button ON Pygame Window

BLIT Suggest\_Button ON Pygame Window

WHILE running IS True DO

IF Checkmate EQUAL TO True DO

IF Load\_Button IS clicked DO

CALL Load\_game()

ENDIF

IF Save\_Button IS clicked DO

CALL Save\_game()

ENDIF

PL\_Count = 0

IF PL\_Count = 0 DO

Crnt\_plyr = “White”

PL\_Count = 1

ELSE DO

Crnt\_plyr = “Black”

PL\_Count = 0

ENDIF

CALL move()

ELSE DO

OUTPUT (“Game Over” + Crnt\_plyr+”Wins”)

ENDIF

ENDWHILE

END

**Load function**

IF file EXISTS DO

OPEN file IN READ MODE

Crnt\_plyr IS FIRST LINE OF file

Board\_pos IS SECOND LINE OF file

CLOSE file

ELSE DO

OUTPUT ‘No save file’

ENDIF

END

**Save Function**

OPEN file IN WRITE MODE

WRITE Crnt\_plyr IN file

WRITE Board\_pos IN file

CLOSE file

END

**Move Function**

Checkmate = False

INPUT Selected\_piece

WHILE piece IS NOT Crnt\_plyr DO

INPUT Selected\_piece

ENDWHILE

IF piece\_destination EQUAL TO king\_pos DO

CALL Check()

IF Checkmate EQUALS True DO

RETURN

ELSE DO

FIND Possible\_moves FOR piece

List = Possible\_moves

HIGHLIGHT List[0:len(List)-1] ON BOARD

INPUT Piece\_destination

DELETE Board\_Pos[Piece]

Piece = Piece\_destination

IF Piece IN Promote\_pos DO

CALL Promotion()

ENDIF

RETURN

ENDIF

END

**Check Function**

Selected\_piece = King

FOR i=0 TO 8 DO

FOR j = 0 TO 8 DO

IF Board\_pos[ i ][ j ] IS Crnt\_side[0] DO //The Current Side’s king

King\_pos = Board\_pos[ i ][ j ]

BREAK

ENDIF

ENDLOOP

ENDLOOP

FOR i=0 TO 8 DO

FOR j = 0 TO 8 DO

IF Board\_pos[ i ][ j ] IS NOT 0 AND Board\_pos[ i ][ j ] IS NOT Crnt\_plyr DO

RETRIEVE moves FOR Board\_pos[ i ][ j ]

FOR k =0 TO LENGTH OF List DO

IF List[ k ] IS IN moves DO

Check = True

RETURN

ENDIF

ENDLOOP

ENDIF

ENDLOOP

ENDLOOP

END

**Promotion Function**

INPUT New\_piece

Board\_pos[ y ][ x ] = New\_piece

END