

Chess Project

Overview

A learning project of Chess game using C programming language without Graphical interface.

we depend on this project on the principal of assumption as the king has 8 moves and we checked the validity of each move.

-Description

-a two player chess game played on a chessboard Using keyboard

-Each player begins the game with 16 pieces: one king, one queen, two rooks, two knights, two bishops, and eight pawns

-Capital ones mean white and small one is black

-Features

-black and white squares are represented by "." for White squares and "-" for Black squares.

-Black pieces are represented by "p","r","n","b","q","k" for pawns, rooks, knights, bishops, queen and king respectively.

White pieces are represented by the same letters but capitalized

-displayed letters from A to H representing columns ,and 1 to 8 representing rows

- Press "s" for saving and continue playing and press "l" for load and 'u' for undo and 'r' for redo
- there is castling which replace the place of rook with the king at either of its sides
- display the captured pieces at every turn
- The game is won by checkmate
- The game may end in a draw by stalemate
- If there is a checkmate there will no valid move except the move with protect the king or prevent the checkmate

How to play

- Read a move from player1 (white) using keyboard ex (b2b3)
- Check if it's a valid move
- If not valid, will print "you can't do this move" and will ask to enter a valid move again
- If there is a checkmate will print "king is under attack" and there will be no valid move except the one to prevent checkmate for king and other pieces
- Then Read moves from player2 (Black)

Description of data structures & functions.

-main()

Calling board function and then enter a move then check every time if this move make checkmate function =0 then its valid and do the move else don't do the move , if turn flag =1 its white turn and if =0 its black

-play()

Four inputs checking if the first input is (s or l) for save and load and then do conversion for letters because board is 2d array start with [0][0] at top left then checking validity for every every piece if any one of them true then it move and old position return "-" or "." If any of them false repeat the scanf

-check_king()

It's the checkmate function which check if there is an a valid move to capture king if there is so its flag turn to 1(r=1) which prevent the move it then it only accept the move which prevent the checkmate only

-stalemate()

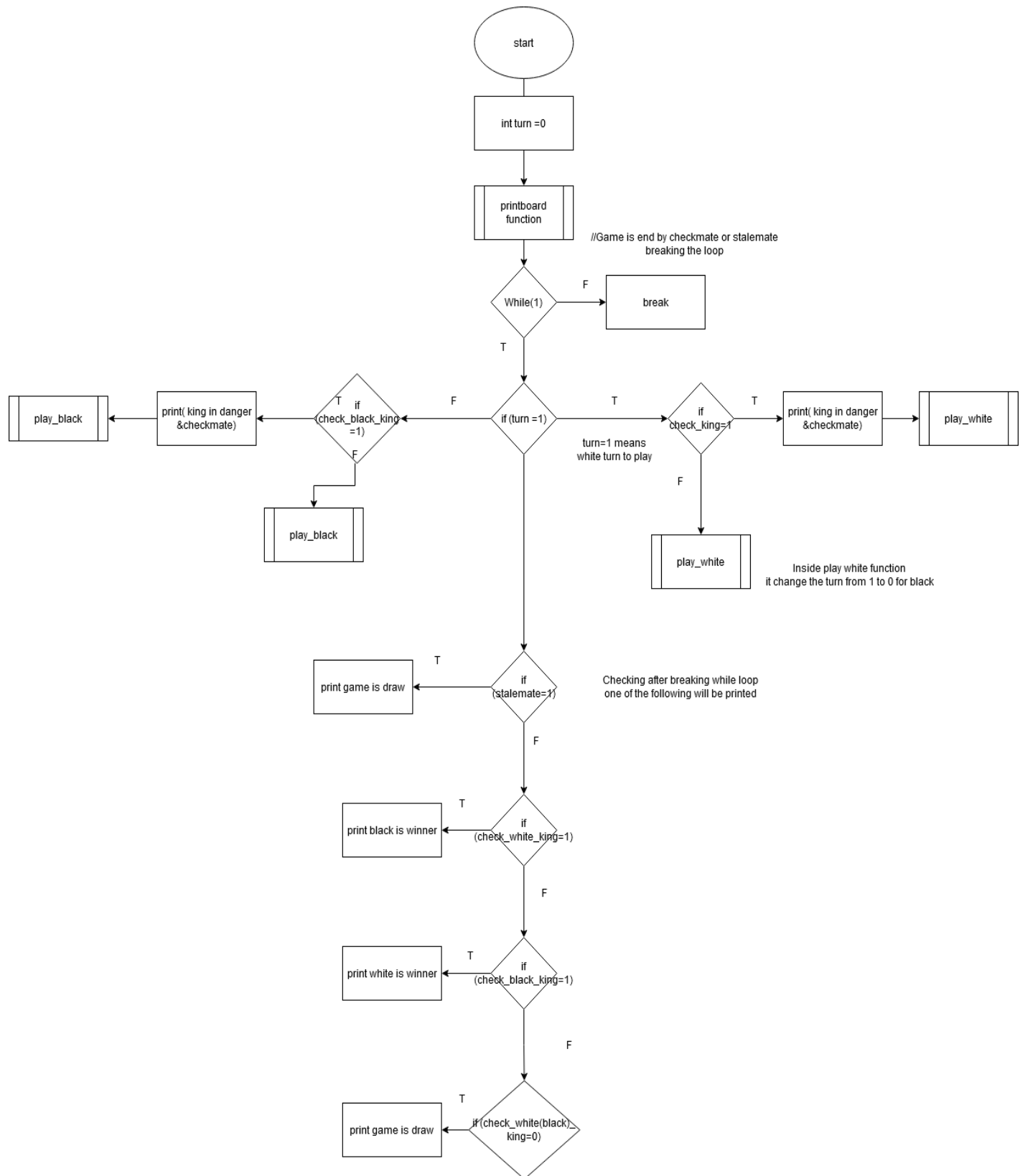
It happened if there is a (king vs king)or (king vs king,knight) or (king vs king ,Bishop) or (king vs king ,2 knights) ... or A king in a safe position and there is not any valid move for king and other pieces.

-end_game()

If there is no move for the king or any other piece to protect the king and prevent the checkmate it will end the game printing the winner and the king is checked.

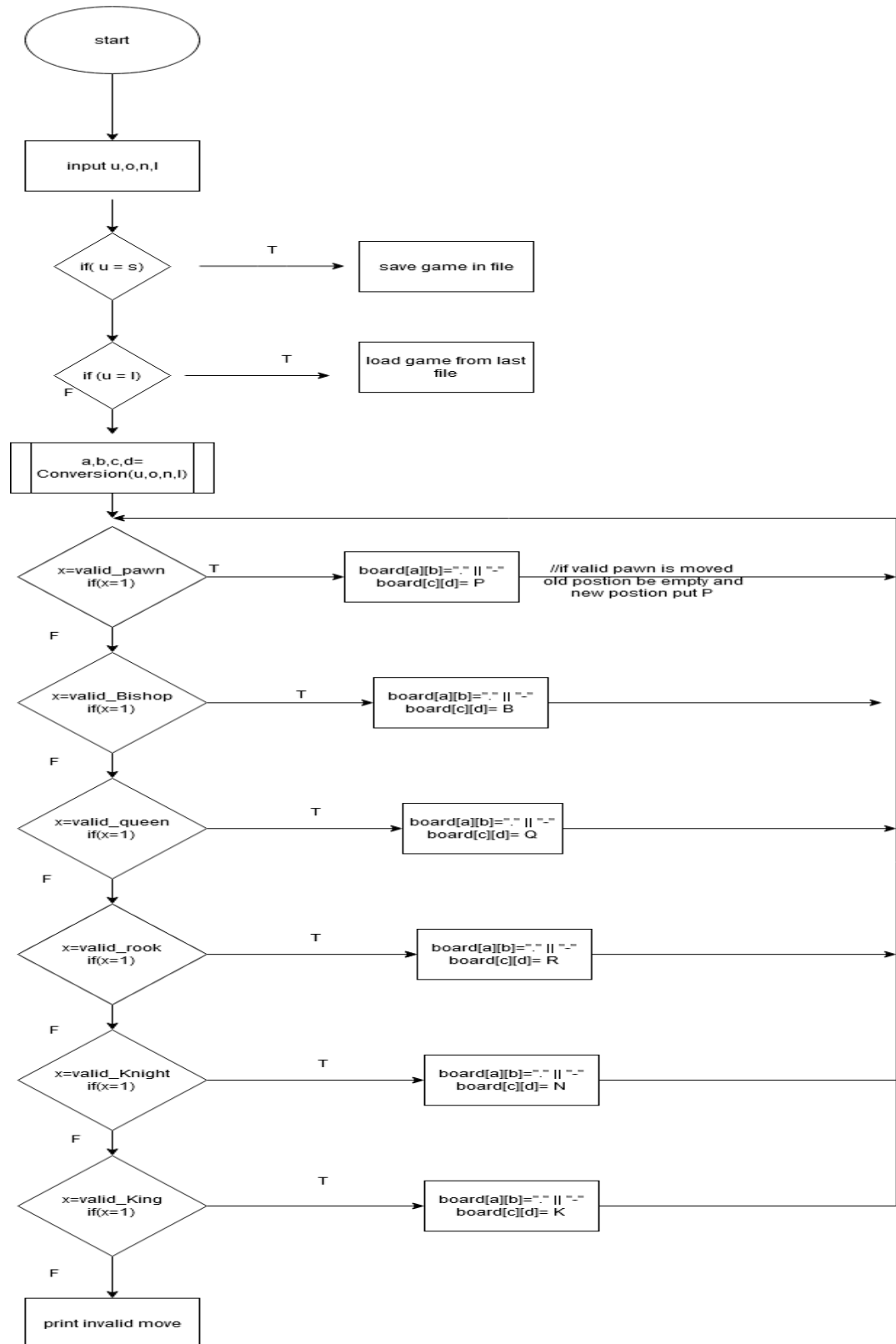
Flowchart

Main()



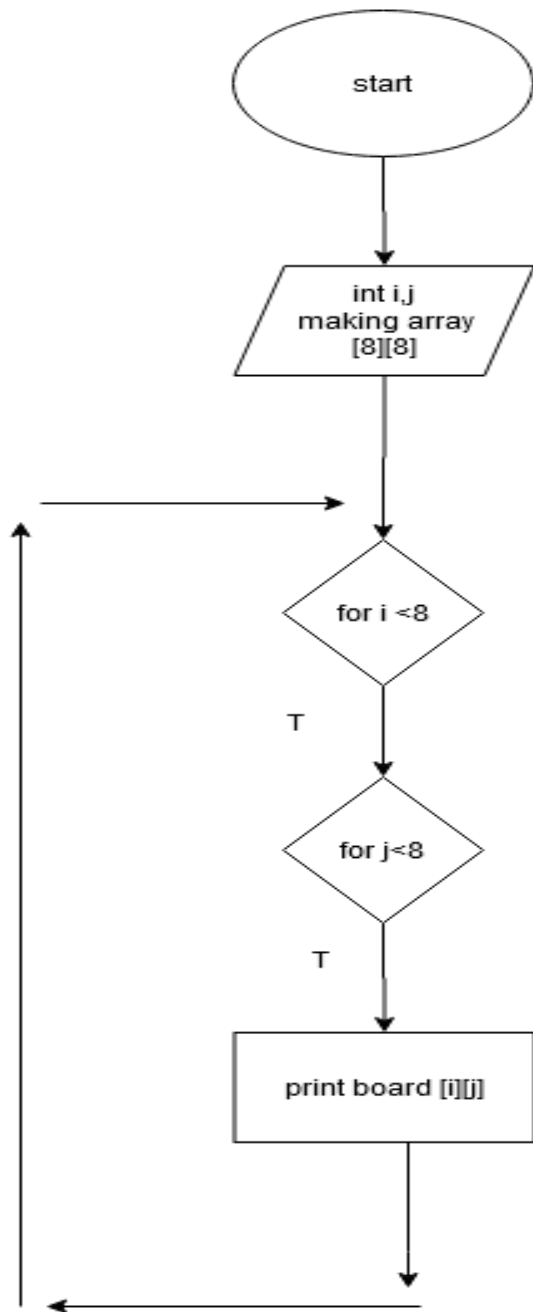
-For play()

Play function



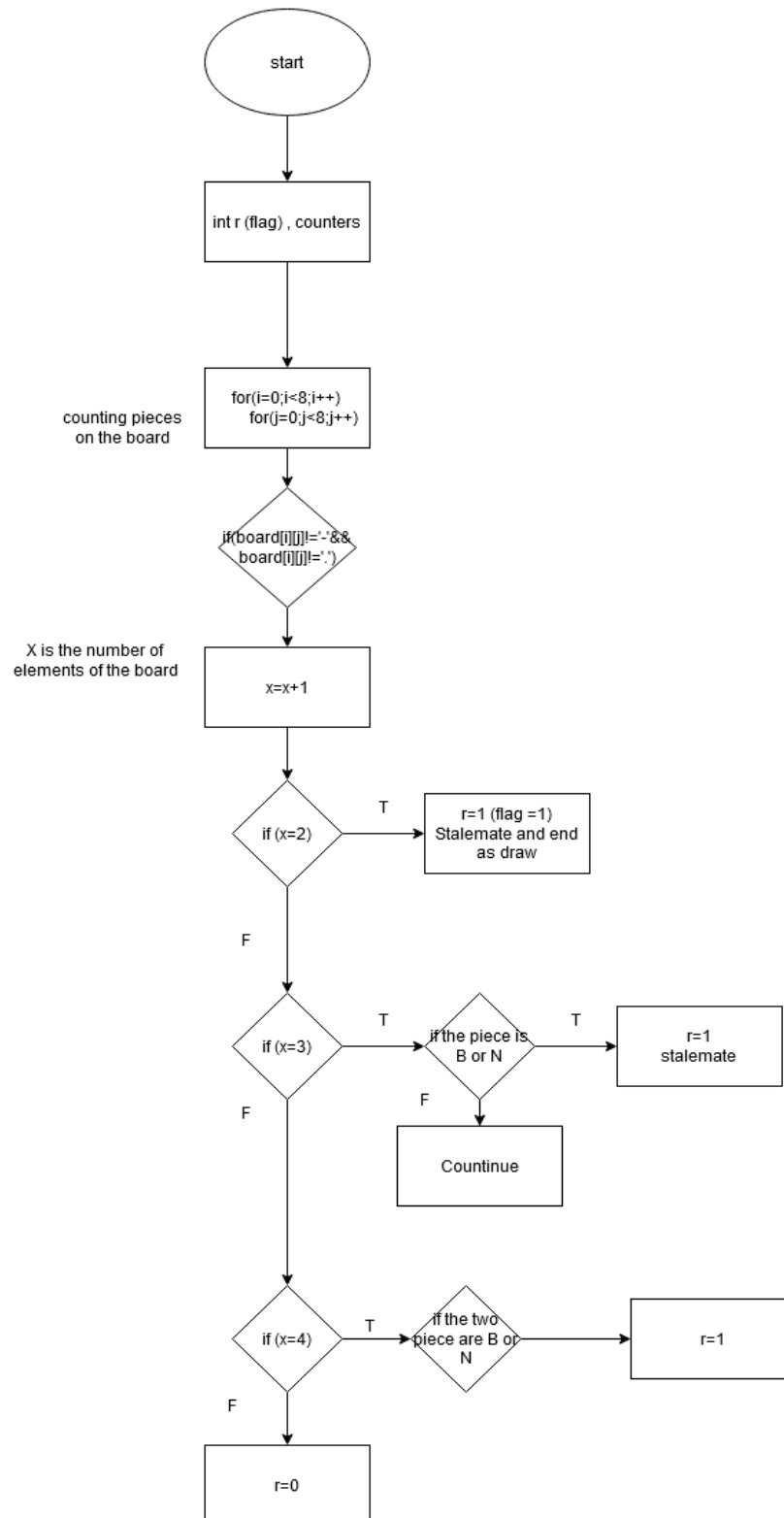
For board

Board Function



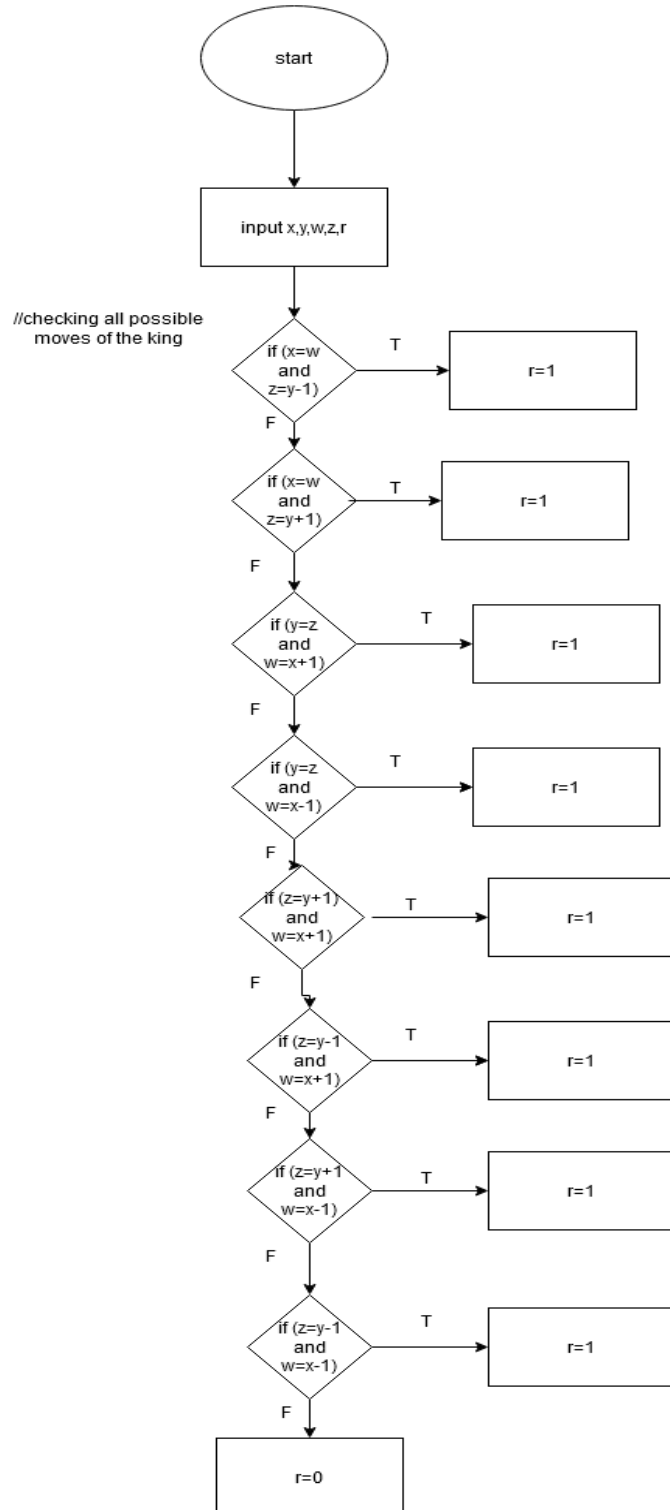
Stalemate ()

Stalemate function



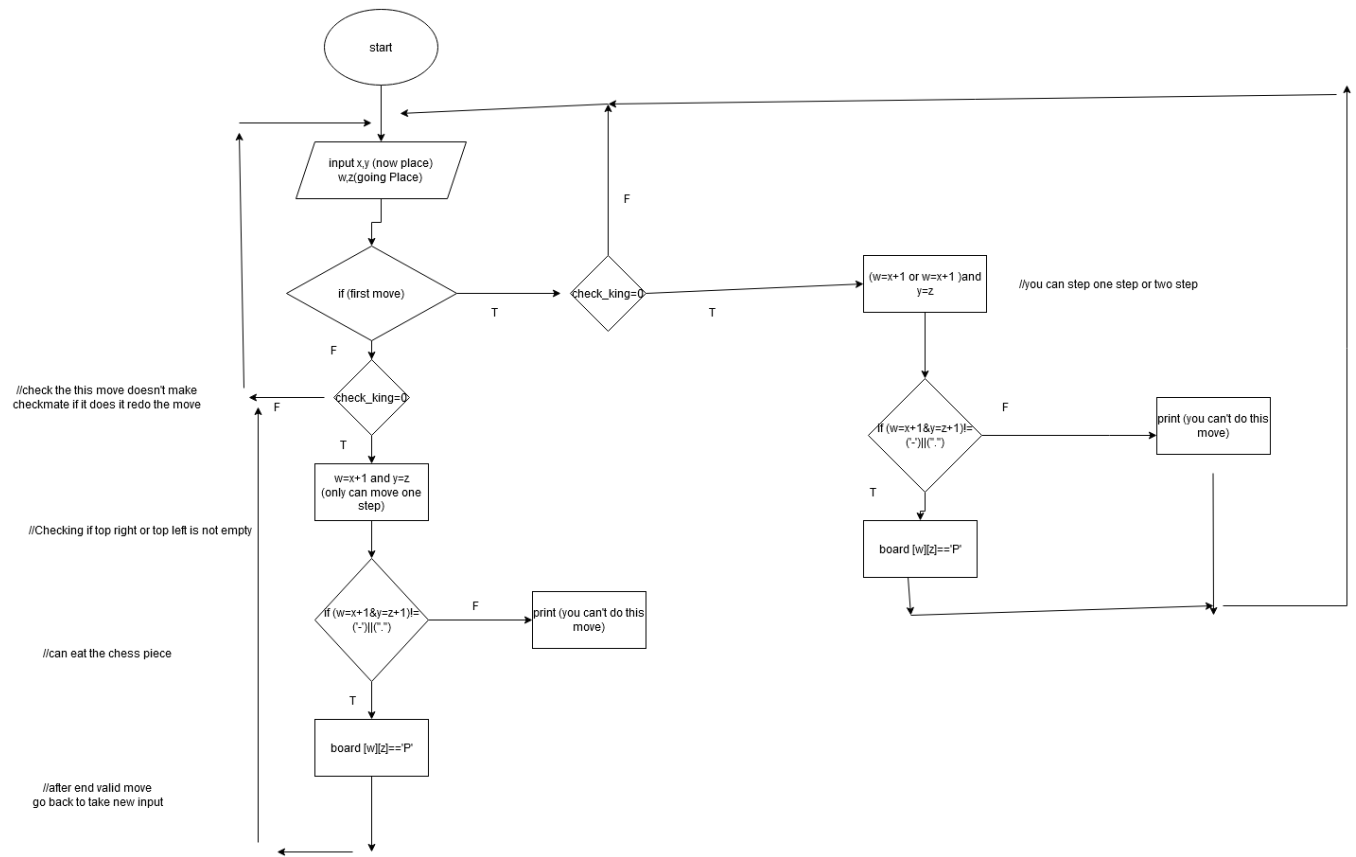
Valid of king()

Valid king function



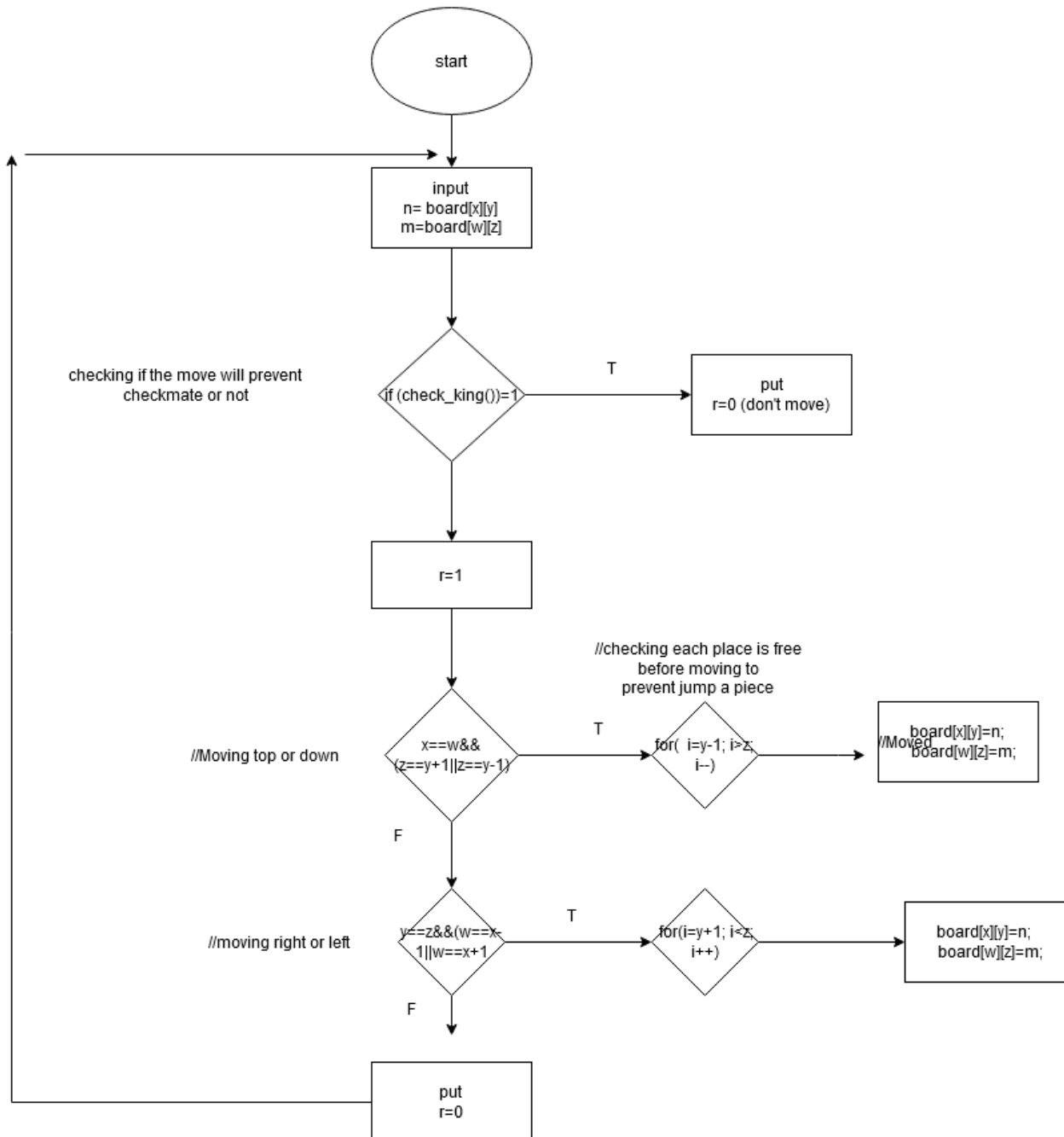
Pawn valid()

Pawn Validity Function



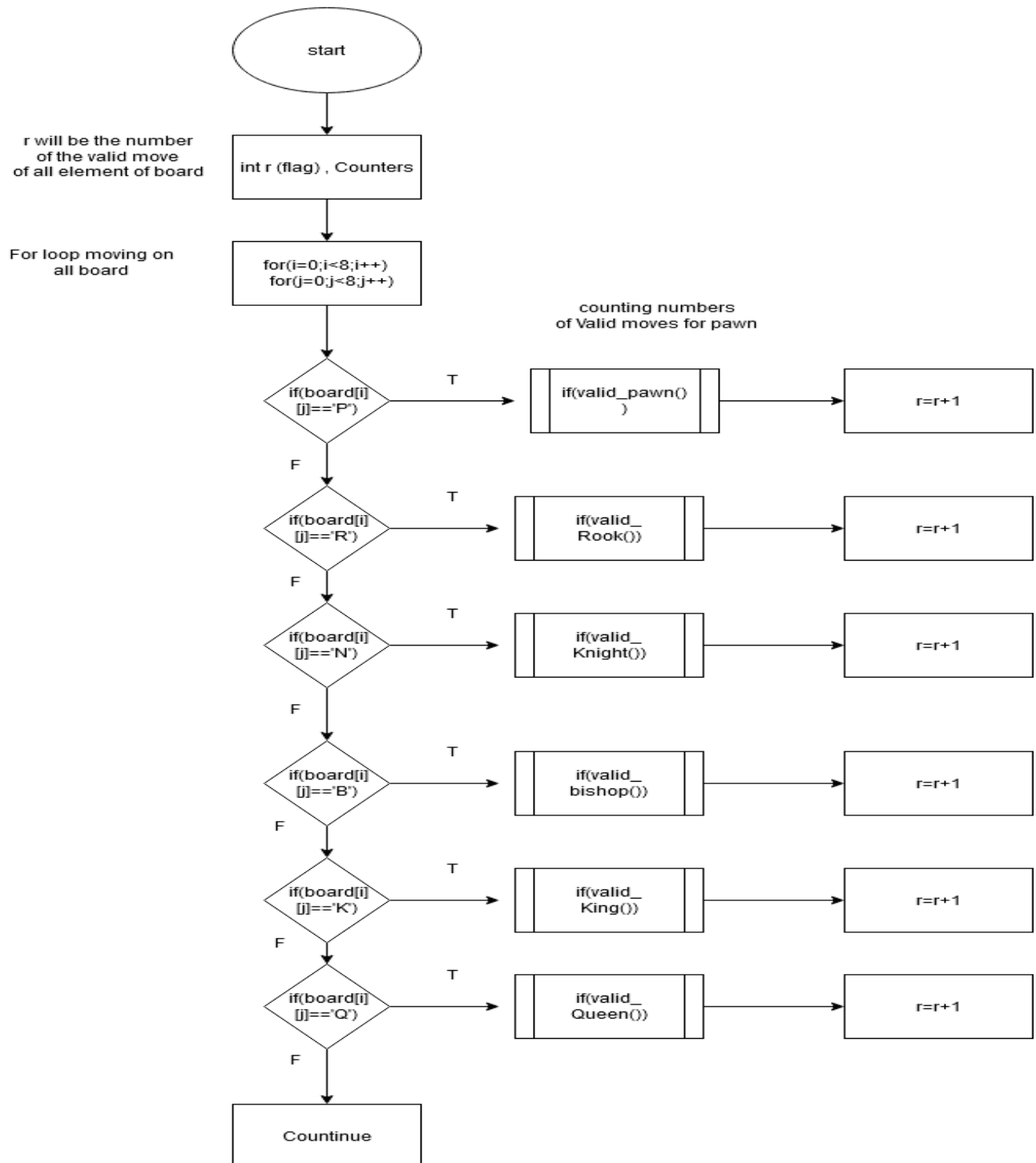
rook valid()

Rook valid fuction



End_game()

Endgame flow



N.B: these flowcharts for the main algorithms in the code source and they made for white pieces, black pieces are same but just changing color (white=>Black)

Pseudocode

main()

- print the first board.
- while loop with condition true for the game.
- check the stalemate and it should be true to continue otherwise it will break the loop.
- Check the turn & check the Checkmate
- Call play function
- Checking end game function=0 or break loop otherwise
- After breaking loop from(Stalemate or Endgame function):
 - Print draw in case of stalemate=1 or
 - check_black_king(white)=0
 - print black is winner in case of check_white_king=1
 - print white is winner in case of check_black_king=1.

Board()

- Making 2d array which is the chess board which start from top left[0][0]
- at which the char. Inputs are conversed to letter using function "Conversion1" which change letters to numbers using else if similar to switch
- and integer inputs also conversion using "Conversion2" and these conversion are stored at global variables named a,b,c,d

Play()

- scan four variable
- checking if first variable is "s ","l", "u", "r "for save , load ,undo , redo respectively
- for castling we check that K and R didn't move and empty place between them
- we check validity of the move First for the chosen piece if it true, then if the going place(board [c][d]) is empty we make old place is "-" or "." And put new place by chosen piece, if the going place isn't empty , check validity then put the old piece on the new 1 and print the eaten pieces
- if pawn went to end if map will be upgraded to one of the following R,Q,B,N

Valid_white_king()

- in validity we depend of the concept of assumption for example the king have 8moves if the user enter another 1 from the eight move will print invalid and enter again and also with other pieces the same concept

Stalemate()

- make flag r and counters of type intger
- making two for loops to count all pieces on the board
- if there are 2 pieces (2kings(K,k)) then r=1 and match end as draw
- if there are 3pieces (2kings and (one horse or one bishop)) then r=1 and match end as draw (stalemate)
- if there are 4 pieces (2kings and two horses togheter at any side) then r=1

end game()

- make a variable i,j (counters), r (count total number of valid move of all pieces)
- making 2 for loops to check every element on the board and another 2 for loops to count the number of valid moves of each element and count it on the variable r
- if the variable r is equal to zero it mean that there is no more valid move at all pieces which lead to end the game and print the winner

Sample run & Screenshots.

1)normal game

```

"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
7 p p p . p p p p 7
6 . . . . . . . 6
5 . . . p . . . 5
4 . . . P . . . 4
3 . . . . . . . 3
2 p p p . p p p p 2
1 R N B Q K B N R 1
-----
a b c d e f g h
white kills :
black kills :
white turn
c1g5
-----
a b c d e f g h
8 r n b q k b n r 8
7 p p p . p p p p 7
6 . . . . . . . 6
5 . . . p . . . B . 5
4 . . . P . . . . 4
3 . . . . . . . 3
2 p p p . p p p p 2
1 R N . Q K B N R 1
-----
a b c d e f g h
white kills :
black kills :
black turn
g8f6
-----
a b c d e f g h

```

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
black kills :
white turn
b1a3
  a b c d e f g h
-----
8 r n b q k b . r 8
7 p p p . p p p p 7
6 . . . . n . . 6
5 . . p . . B . 5
4 . . P . . . . 4
3 N . . . . . 3
2 p p p . p p p p 2
1 R . . Q K B N R 1
-----
  a b c d e f g h
white kills :
black kills :
black turn
```

2)save

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
2 . P P P P P P P 2
1 R N B Q K B N R 1
-----
  a b c d e f g h
white kills :
black kills :
black turn
F7f6
  a b c d e f g h
-----
8 r n b q k b n r 8
7 p p p p p . p p 7
6 . . . . p . . 6
5 . . . . . . 5
4 P . . . . . 4
3 . . . . . . 3
2 . P P P P P P P 2
1 R N B Q K B N R 1
-----
  a b c d e f g h
white kills :
black kills :
white turn
5
Save Succesful
Countinue Playing
white turn
```

3)load after closing and opening

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"

  a b c d e f g h
-----
8 r n b q k b n r 8
7 p p p p p p p 7
6 . . . . . 6
5 . . . . . 5
4 . . . . . 4
3 . . . . . 3
2 P P P P P P P 2
1 R N B Q K B N R 1
-----
  a b c d e f g h
white turn
1
  a b c d e f g h
-----
8 r n b q k b n r 8
7 p p p p p p p 7
6 . . . . . p . 6
5 . . . . . 5
4 P . . . . . 4
3 . . . . . 3
2 . P P P P P P P 2
1 R N B Q K B N R 1
-----
  a b c d e f g h
Load is Successful
white turn
e
5
```

4)castling

-Castling at lower left

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"

b8c6
  a b c d e f g h
-----
8 r . . . k b n r 8
7 . . . . p p p p 7
6 b p n . . . . 6
5 p . P . q . . 5
4 P . . p . . N 4
3 . . N . B . . 3
2 . P P Q P P P P 2
1 R . . . K B - R 1
-----
  a b c d e f g h
white kills :
black kills :
white turn
e1c1
  a b c d e f g h
-----
8 r . . . k b n r 8
7 . . . . p p p p 7
6 b p n . . . . 6
5 p . P . q . . 5
4 P . . p . . N 4
3 . . N . B . . 3
2 . P P Q P P P P 2
1 . . K R - B - R 1
-----
  a b c d e f g h
white kills :
```


-Castling at upper left

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
8 r - . - k b n r 8
7 - . - . p p p p 7
6 b p n - . - . 6
5 p . P . q . - . 5
4 P - . p . - . N 4
3 - . N . B . - . 3
2 . P P Q P P P P 2
1 - . K R - B - R 1
-----
a b c d e f g h
white kills :
black kills :
black turn
e8c8
-----
a b c d e f g h
8 - . k r . b n r 8
7 - . - . p p p p 7
6 b p n - . - . 6
5 p . P . q . - . 5
4 P - . p . - . N 4
3 - . N . B . - . 3
2 . P P Q P P P P 2
1 - . K R - B - R 1
-----
a b c d e f g h
white kills :
black kills :
black turn
```

-Castling at upper right

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
h1h2
-----
a b c d e f g h
8 r n . - k - . r 8
7 - . - . p p p p 7
6 b p . - . n . - 6
5 p . P . q . b . 5
4 P - . p . n . b 4
3 - . N . B N - P 3
2 . P P Q P P P R 2
1 R . - . K B - . 1
-----
a b c d e f g h
white kills :
black kills :
black turn
e8g8
-----
a b c d e f g h
8 r n . - . r k - 8
7 - . - . p p p p 7
6 b p . - . n . - 6
5 p . P . q . b . 5
4 P - . p . n . b 4
3 - . N . B N - P 3
2 . P P Q P P P R 2
1 R . - . K B - . 1
-----
a b c d e f g h
white kills :
```

5)checkmate

moving the queen which make checkmate on the king and there will be no moves allowed except the queen move from e7 to f7 to protect the king and prevent the checkmate

1-moving queen to make checkmate

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
-----
8 r n b - k b n r 8
7 p . p p q . p p 7
6 . - . - . - . - 6
5 - p - . p . . 5
4 . - P - . - Q - 4
3 - . . . . P . 3
2 . - . P . P . P 2
1 R N B . K B N R 1
-----
a b c d e f g h
white turn
g4h5
a b c d e f g h
-----
8 r n b - k b n r 8
7 p . p p q . p p 7
6 . - . - . - . - 6
5 - p - . p . - Q 5
4 . - P - . - . - 4
3 - . . . . P . 3
2 . - . P . P . P 2
1 R N B . K B N R 1
-----
a b c d e f g h
white kills :
black kills :
black king is under attack
black turn
```

2-trying wrong moves which doesn't protect the king

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
5 - p - . p . - . 5
4 . - P - . - Q - 4
3 - . - . - . P . 3
2 . - . P . P . P 2
1 R N B . K B N R 1
-----
a b c d e f g h
white turn
g4h5
a b c d e f g h
-----
8 r n b - k b n r 8
7 p . p p q . p p 7
6 . - . - . - . - 6
5 - p - . p . - Q 5
4 . - P - . - . - 4
3 - . - . - . P . 3
2 . - . P . P . P 2
1 R N B . K B N R 1
-----
a b c d e f g h
white kills :
black kills :
black king is under attack
black turn
e7e6
you can't do this move
black king is under attack
black turn
```

3-printing only the valid move

```
Select "C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
1 R N B . K B N R 1
-----
a b c d e f g h
white turn
g4h5
a b c d e f g h
-----
8 r n b - k b n r 8
7 p . p p q . p p 7
6 . - . - . - . - 6
5 - p - . p . - Q 5
4 . - P - . - . - 4
3 - . - . - . P . 3
2 . - . P . P . P 2
1 R N B . K B N R 1
-----
a b c d e f g h
white kills :
black kills :
black king is under attack
black turn
e7e6
you can't do this move
black king is under attack
black turn
g8h6
you can't do this move
black king is under attack
black turn
```

```
Select "C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
a b c d e f g h
white kills :
black kills :
black king is under attack
black turn
e7e6
you can't do this move
black king is under attack
black turn
g8h6
you can't do this move
black king is under attack
black turn
e7f7
a b c d e f g h
-----
8 r n b - k b n r 8
7 p . p p - q p p 7
6 . . . . . 6
5 - p - . p . - Q 5
4 . - P - . . - 4
3 . . . . . P . 3
2 . . . P . P . P 2
1 R N B . K B N R 1
-----
a b c d e f g h
white kills :
black kills :
white turn
```

6)Upgrading pawn when reaching at end of map

"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"

```
  a b c d e f g h
-----
8  . . . - k b n r 8
7  - P - . - . p p 7
6  . - . - . - . - 6
5  - . - . - . - . 5
4  . - . - . - . - 4
3  - . - . - . - . 3
2  . - . - . - . - 2
1  R N - . K B N R 1
```

```
  a b c d e f g h
```

white turn

b7b8

```
  a b c d e f g h
-----
8  . P . - k b n r 8
7  - . - . - . p p 7
6  . - . - . - . - 6
5  - . - . - . - . 5
4  . - . - . - . - 4
3  - . - . - . - . 3
2  . - . - . - . - 2
1  R N - . K B N R 1
```

```
  a b c d e f g h
```

white kills :

black kills :

Choose one of the following R,Q,B,N

Q

```
  a b c d e f g h
-----
```

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"

-----
8 . P . - k b n r 8
7 - . - . - . p p 7
6 . - . - . - . - 6
5 - . - . - . - . 5
4 . - . - . - . - 4
3 - . - . - . - . 3
2 . - . - . - . - 2
1 R N - . K B N R 1
-----
a b c d e f g h
white kills :
black kills :
Choose one of the following R,Q,B,N
Q
a b c d e f g h
-----
8 . Q . - k b n r 8
7 - . - . - . p p 7
6 . - . - . - . - 6
5 - . - . - . - . 5
4 . - . - . - . - 4
3 - . - . - . - . 3
2 . - . - . - . - 2
1 R N - . K B N R 1
-----
a b c d e f g h
black king is under attack
black turn
-
```

6)stalemate

-case of 2 kings

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
  a b c d e f g h
-----
8 . . . - k - . - 8
7 - . . . - . - . 7
6 . . . . - . - . 6
5 - . . . - . - . 5
4 . . . . - . - . 4
3 - . . . - . - . 3
2 . . . - p - . - 2
1 - . . . K - . - 1
-----
  a b c d e f g h
white turn
e1e2
  a b c d e f g h
-----
8 . . . - k - . - 8
7 - . . . - . - . 7
6 . . . . - . - . 6
5 - . . . - . - . 5
4 . . . . - . - . 4
3 - . . . - . - . 3
2 . . . - K - . - 2
1 - . . . - . - . 1
-----
  a b c d e f g h
white kills :
black kills : p
game is draw!
Process returned 0 (0x0)   execution time : 8.519 s
Press any key to continue.
```

-in case of 2king and 1horse

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
8 . . . - k - . - 8
7 - . . . - n - . 7
6 . . . . - . - . 6
5 - . . . - . - . 5
4 . . . . - . - . 4
3 - . . . - . - . 3
2 . . . - p - . - 2
1 - . . . K - . - 1
-----
  a b c d e f g h
white turn
e1e2
  a b c d e f g h
-----
8 . . . - k - . - 8
7 - . . . - n - . 7
6 . . . . - . - . 6
5 - . . . - . - . 5
4 . . . . - . - . 4
3 - . . . - . - . 3
2 . . . - K - . - 2
1 - . . . - . - . 1
-----
  a b c d e f g h
white kills :
black kills : p
game is draw!
Process returned 0 (0x0)   execution time : 6.681 s
Press any key to continue.
```

-in case of 2 kings and 2 knights at the same side

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
8 . . . . k . . . 8
7 . . n . . n . . 7
6 . . . . . . . . 6
5 . . . . . . . . 5
4 . . . . . . . . 4
3 . . . . . . . . 3
2 . . . . p . . . 2
1 . . . . K . . . 1
-----
  a b c d e f g h
white turn
e1e2
  a b c d e f g h
-----
8 . . . . k . . . 8
7 . . n . . n . . 7
6 . . . . . . . . 6
5 . . . . . . . . 5
4 . . . . . . . . 4
3 . . . . . . . . 3
2 . . . . K . . . 2
1 . . . . . . . . 1
-----
  a b c d e f g h
white kills :
black kills : p
game is draw!
Process returned 0 (0x0)   execution time : 16.726 s
Press any key to continue.
```

-in case of 2 kings and one bishop

```
"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"
8 . . . . k . . . 8
7 . . . . B . . . 7
6 . . . . . . . . 6
5 . . . . . . . . 5
4 . . . . . . . . 4
3 . . . . . . . . 3
2 . . . . . . . . 2
1 . . . . K . . B 1
-----
  a b c d e f g h
white kills :
black kills :
black turn
e8e7
  a b c d e f g h
-----
8 . . . . . . . . 8
7 . . . . k . . . 7
6 . . . . . . . . 6
5 . . . . . . . . 5
4 . . . . . . . . 4
3 . . . . . . . . 3
2 . . . . . . . . 2
1 . . . . K . . B 1
-----
  a b c d e f g h
white kills : B
black kills :
game is draw!
```


-Stalemate when the king is checkmate from all position

"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"

```
a b c d e f g h
-----
8 . . . . . . . 8
7 . . . . . . . 7
6 . p . . . . . 6
5 - P - K - . . . 5
4 . . . . . . . 4
3 . . . . . . . 3
2 . . . N . p . . 2
1 - . . . - B - k 1
-----
a b c d e f g h
white turn
d2f3
a b c d e f g h
-----
8 . . . . . . . 8
7 . . . . . . . 7
6 . p . . . . . 6
5 - P - K - . . . 5
4 . . . . . . . 4
3 . . . . . N . . 3
2 . . . . . p . . 2
1 - . . . - B - k 1
-----
a b c d e f g h
white kills :
black kills :
black turn
h1g1
you can't do this move
game is draw//
Process returned 0 (0x0)   execution time : 21.504 s
Press any key to continue.
```

7)End game

"C:\Users\USER\Desktop\Kolya\Programing\Chess test 1\bin\Debug\Chess test 1.exe"

```
  a b c d e f g h
-----
8 . n . - k - . - 8
7 R . . . . . 7
6 . - . . . . r 6
5 - . . . . . Q . 5
4 . P p - . . . - 4
3 - . . . . . . 3
2 . - . . . . . 2
1 - . . . K . . . 1
```

```
  a b c d e f g h
```

white turn

g5e7

```
  a b c d e f g h
-----
8 . n . - k - . - 8
7 R . . . Q . . . 7
6 . - . . . . r 6
5 - . . . . . . 5
4 . P p - . . . - 4
3 - . . . . . . 3
2 . - . . . . . 2
1 - . . . K . . . 1
```

```
  a b c d e f g h
```

white kills :

black kills :

black king is under attack

black turn

e8f8

you can't do this move

White is winner

Process returned 0 (0x0) execution time : 78.693 s

Press any key to continue.