User Manual

command-line-based Jungle game

Group 104, Fall 2025

Starting screen

```
Enter command: start | load | replay | exit
```

Users can type any of the following commands to perform corresponding actions.

Start: starts a new game.

Load: loads a saved game (.jungle) to continue playing.

Replay: replays the past record of a game (.record).

Exit: exits the program.

Upon starting a new game, users are prompted to enter names for each player of sides of Red and Blue. If left empty, the player will be given a random name. Note that both players could share the same name only in manual input, but not when left empty for random names.

```
Enter Red player name (leave empty for random): tom
Enter Blue player name (leave empty for random):
```

```
Enter Red player name (leave empty for random): jerry
Enter Blue player name (leave empty for random): jerry
```

Example of random names:

```
Enter Red player name (leave empty for random):
Enter Blue player name (leave empty for random):
Current player: Tiger (RED)
Next player: Dragon (BLUE)
```

A 7x9 game board is now initialized on screen with animal pieces on the starting squares, traps, rivers and dens. An animal piece is represented in the form of player color + animal rank as number as shown in Table 1. For instance, R8 is the Elephant piece of player Red.

Unoccupied squares are shown as ".", river squares as "~~", trap squares as "[ ]", and den squares as "DD". When an animal piece is moved to a specific square, the text of the square is replaced by that of the piece.

The text below also shows the current game states, including the remaining pieces and undos of each player, as well as the current and next player while they take turns moving their pieces.

```
   A  B  C  D  E  F  G
1 B7  . [] DD []  . B6
2  . B3  . []  . B2  .
3 B1  . B5  . B4  . B8
4  . ~~ ~~  . ~~ ~~  .
5  . ~~ ~~  . ~~ ~~  .
6  . ~~ ~~  . ~~ ~~  .
7 R8  . R4  . R5  . R1
8  . R2  . []  . R3  .
9 R6  . [] DD []  . R7
Remaining pieces: tom (Red): 8, Panda (Blue): 8
Remaining undo: tom (Red): 3, Panda (Blue): 3
Current player: tom (RED)
Next player: Panda (BLUE)
```

Table 1: Pieces and their ranks.

Rank	Piece (en)	Piece (cn)
8	Elephant	象
7	Lion	獅
6	Tiger	虎
5	Leopard	豹
4	Wolf	狼
3	Dog	狗
2	Cat	貓
1	Rat	鼠

Basic gameplay

Players can choose to perform move a piece, undo, save, or stop in this line in their turn. A player can perform any possible moves in the form of move [source position] [destination position], for instance, “move A7 A6” causes the piece to move as follows.

```
Enter command (move [srcPos] [desPos] | undo | save | stop): move A7 A6
```

Players would receive a warning text if they have input invalid commands as shown in following:

- If there is no piece at the source position:

```
Enter command (move [srcPos] [desPos] | undo | save | stop): move B1 B2
No piece at the source position.
```

- If the move format is invalid:

```
Enter command (move [srcPos] [desPos] | undo | save | stop): move
Invalid move format. Example: move A7 A6
```

- If the command is unknown:

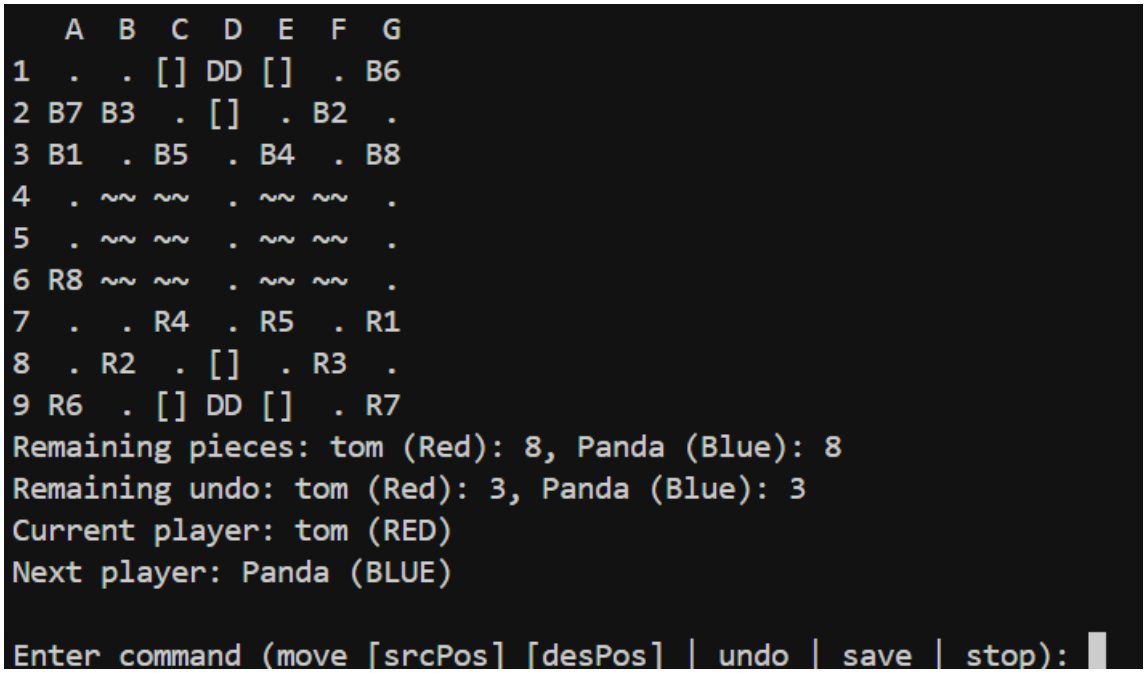
```
Enter command (move [srcPos] [desPos] | undo | save | stop): aaa
Unknown command.
```

Note that the spaces in input commands are trimmed, and commands with excessive spaces are also considered valid.

All commands, except names, are case-insensitive.

```
move      A1 A2
```

Upon successful move action, the updated game board is printed on screen so that the opponent can perform their action.

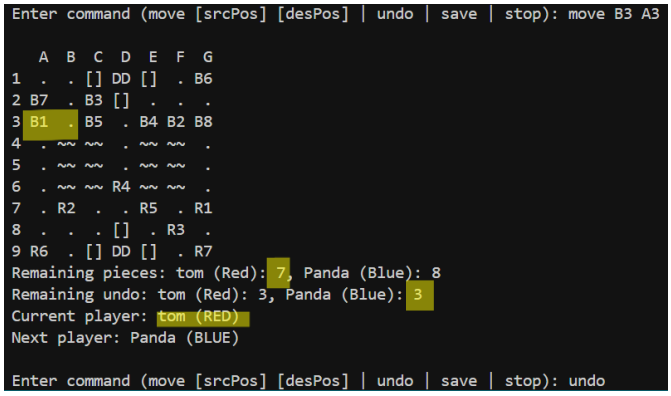


Undo

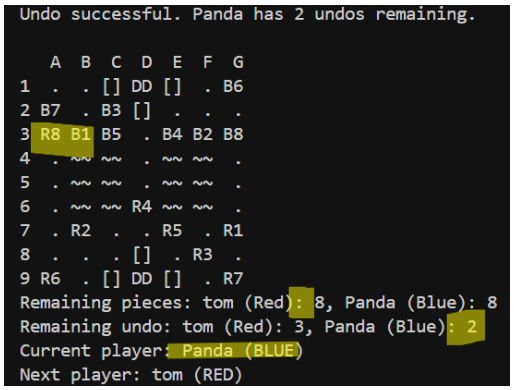
Users can undo their moves if they still have remaining undos, which is maximum 3 times for each player. In the opponent's turn and before the opponent makes a move, the player can input "undo" to revert before the previous move was made. The turn would switch alternatively if multiple "undo" is input.

Here is an example of performing "undo":

- If Red player requests to undo his/her move, the "undo" input is entered under Blue's turn, i.e. when Blue is the current active player.
- The method returns the most recent state before Red moves.
- The remaining undo count for Red player would reduce by one.
- Red player becomes the current active player.



Before undo



After undo

The game will show the remaining undos after successful action.

```
Enter command (move [srcPos] [desPos] | undo | save | stop): undo
Undo successful. Panda has 0 undos remaining.
```

When a player tries to undo their move after using all undo attempts, the undo action will become unsuccessful:

```
Enter command (move [srcPos] [desPos] | undo | save | stop): undo
No undos remaining for Panda.
```

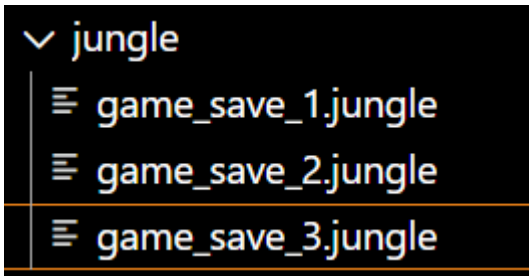
When there are no available moves to undo:

```
Enter command (move [srcPos] [desPos] | undo | save | stop): undo
No moves to undo.
```

Save

Users can save the game manually to the folder jungle in the file format "game_save_ID.jungle", where the ID increments with each saved file. Users can continue playing the game after saving.

```
Enter command (move [srcPos] [desPos] | undo | save | stop): save
Game saved to jungle\game_save_3.jungle
```



If users input IDs with starting zeroes, it will be removed to integer form.

```
Enter command (move [srcPos] [desPos] | undo | save | stop): save
Enter save id to overwrite or new id (01), or leave empty for next: 007
Game saved to jungle\game_save_7.jungle
Game saved.
```

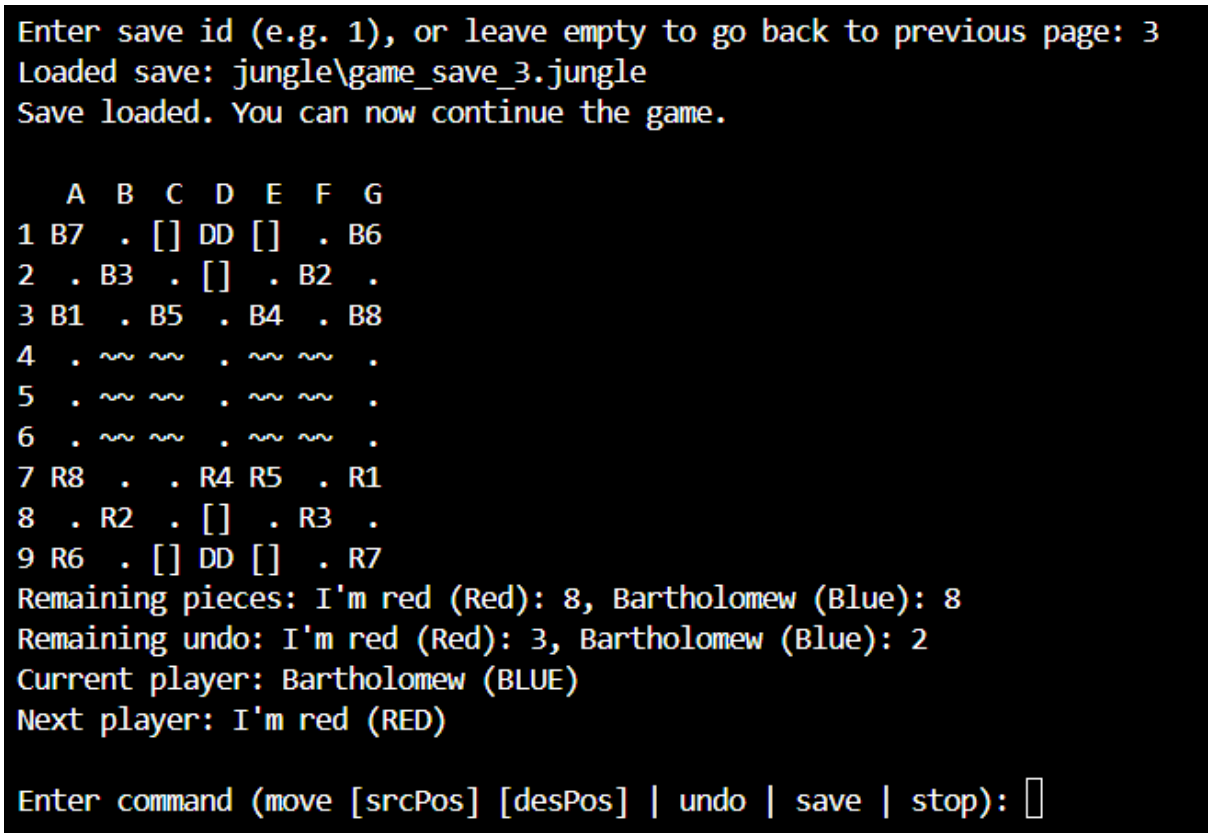
To stop playing the game, input "stop" and the system returns to starting screen:

```
Current player: tom (RED)
Next player: Panda (BLUE)

Enter command (move [srcPos] [desPos] | undo | save | stop): stop
Game stopped.
Enter command: start | load | replay | exit
```

Load

To load a saved .jungle file, users can load it from the starting screen by inputting “load”. Players can continue playing the game after loading the file.



Replay

All game records are saved as .record log files regardless of whether the user has saved the game as a loadable .jungle file. Users can enter replay mode by typing “replay” in the starting screen. The game will then replay the corresponding record. Users cannot amend the chess moves in replay mode.

Records can be replayed in auto mode, or step mode, in which users can cancel replaying only in step mode.

```
Enter record number (e.g. 1, 11), or leave empty to go back to previous page: 11
Attempting to load: record\game_log_11.record
Loaded 17 commands.
Replay mode? (auto | step | cancel):
```

```
Replay mode? (auto | step | cancel): auto
>> start tom Panda

   A  B  C  D  E  F  G
1 B7  . [] DD []  . B6
2  . B3  . []  . B2  .
3 B1  . B5  . B4  . B8
4  . ~~ ~~  . ~~ ~~  .
5  . ~~ ~~  . ~~ ~~  .
6  . ~~ ~~  . ~~ ~~  .
7 R8  . R4  . R5  . R1
8  . R2  . []  . R3  .
9 R6  . [] DD []  . R7
Remaining pieces: tom (Red): 8, Panda (Blue): 8
Remaining undo: tom (Red): 3, Panda (Blue): 3
Current player: tom (RED)
Next player: Panda (BLUE)

>> move A7 A6
A7 -> A6
```

Auto mode

```
Replay mode? (auto | step | cancel): step
>> start tom Panda

   A  B  C  D  E  F  G
1 B7  . [] DD []  . B6
2  . B3  . []  . B2  .
3 B1  . B5  . B4  . B8
4  . ~~ ~~  . ~~ ~~  .
5  . ~~ ~~  . ~~ ~~  .
6  . ~~ ~~  . ~~ ~~  .
7 R8  . R4  . R5  . R1
8  . R2  . []  . R3  .
9 R6  . [] DD []  . R7
Remaining pieces: tom (Red): 8, Panda (Blue): 8
Remaining undo: tom (Red): 3, Panda (Blue): 3
Current player: tom (RED)
Next player: Panda (BLUE)

Press Enter to continue (or 'cancel' to stop replay):
```

Step mode

Winning conditions

- Any piece entering opponent's den

```
Enter command (move [srcPos] [desPos] | undo | save | stop): move d8 d9
nnn (Blue) wins by entering Red's den!
Game over! Winner: nnn (Blue)
```

- Capturing all opponent's pieces

```
Enter command (move [srcPos] [desPos] | undo | save | stop): move d7 e7
Bartholomew (Blue) wins by capturing all opponent's pieces!
Game over! Winner: Bartholomew (Blue)
```