

```
Script started on 2023-12-05 19:00:18+00:00 [TERM="xterm-256color" TTY="/dev/pts/1" COLUMNS="48" LINES="51"]
\[\033[01;34m\]\w\[\033[00m\]$ pwd
/home/runner/Project-8-Double-the-Fun-kcp3s
\[\033[01;34m\]\w\[\033[00m\]$ ls -la
total 2536
drwxr-xr-x 1 runner runner    350 Dec  5 19:00 .
drwxrwxrwx 1 runner runner    112 Dec  5 15:39 ..
-rwxr-xr-x 1 runner runner 22904 Dec  5 17:04 a.out
-rw-r--r-- 1 runner runner    17 Oct 27 20:51 .breakpoints
drwxr-xr-x 1 runner runner    18 Dec  1 01:19 .cache
drwxr-x--- 1 runner runner   414 Dec  1 02:08 .cccls-cache
-rw-r--r-- 1 runner runner  2055 Dec  5 18:57 ChessGameLL.cpp
-rw-r--r-- 1 runner runner   524 Dec  5 18:57 ChessGameLL.h
-rw----- 1 runner runner    13 Dec  1 01:18 foolsmate.dat
-rw----- 1 runner runner   357 Dec  1 01:18 game1.dat
-rw----- 1 runner runner   490 Dec  1 01:18 game2.dat
-rw-r--r-- 1 runner runner    15 Dec  5 16:35 game3.dat
drwxr-xr-x 1 runner runner    68 Dec  5 18:57 .lesson
-rwxr-xr-x 1 runner runner 1254392 Oct 27 20:53 main
-rw-r--r-- 1 runner runner   1107 Dec  5 18:57 main.cpp
-rwxr-xr-x 1 runner runner 1255712 Oct 27 20:53 main-debug
-rw-r--r-- 1 runner runner   449 Oct 27 20:53 Makefile
-rw-r--r-- 1 runner runner     0 Dec  5 19:00 Patel_Project_8.log
-rw-r--r-- 1 runner runner  1426 Dec 21 2022 .replit
-rw-r--r-- 1 runner runner   141 Dec  1 01:19 replit.nix
\[\033[01;34m\]\w\[\033[00m\]$ cat -n main.cpp
 1  #include "ChessGameLL.h"
 2  #include <fstream>
 3  #include <iostream>
 4  #include <string>
 5
 6  int main() {
 7      std::ifstream file;
 8      std::string filename;
 9      ChessGameLL object;
10
11      std::cout << "Enter a game to review: ";
12      std::cin >> filename;
13
14      file.open(filename);
15
16      if (!file) {
17          std::cout << "INVALID FILE\n";
18          return 0;
19      }
20      int counter = 1;
21
22      // TO get read the file using white first element and iterate untill it ends
23      Turn w;
24      while (file >> w.move) {
25          w.id = counter;
26          w.color = 'W';
27          object.push_back(w);
28
29          Turn b;
30          b.id = w.id;
31          b.color = 'B';
32          file >> b.move;
33          if(b.move != ""){
34              object.push_back(b);}
35          counter++;
36      }
37      file.close();
38
39
40      // This uses implementation to get the file right
41      char move;
42      std::cout << "\n---Game Start---\n";
43      object.firstmove();
```

```
44     do {
45         std::cout << "Enter n/p/q: ";
46         std::cin >> move;
47
48         if (move == 'n') {
49             object.next_move();
50         } else if (move == 'p') {
51             object.prev_move();
52         }
53     } while (move == 'n' || move == 'p');
54
55     if (move == 'q') {
56         std::cout << "--- Ending Review ---\n";
57         return 0;
58     }
59 } \[\033[01;34m\]\w\[\033[00m\]$ cat -n ChessGameLL.h
 1 #ifndef _LINKEDLIST_H
 2 #define _LINKEDLIST_H
 3 #include <string>
 4
 5 struct Turn {
 6     std::string move; // algebraic notation string
 7     char color;       // B for black W for White
 8     int id;           // which turn is it
 9 };
10
11 struct Node {
12     Turn data;
13     Node *next;
14     Node *prev;
15 };
16
17 class ChessGameLL {
18 private:
19     Node *head;
20     Node *current_move;
21
22 public:
23     ChessGameLL();
24     ~ChessGameLL();
25
26     bool empty() const;
27     void push_back(Turn turn);
28     void next_move();
29     void prev_move();
30     void firstmove();
31     void winner();
32 };
33
34 #endif \[\033[01;34m\]\w\[\033[00m\]$ cat -n ChessGameLL.cpp
 1 #include "ChessGameLL.h"
 2 #include <iostream>
 3
 4 ChessGameLL::ChessGameLL() {
 5     head = nullptr;
 6     current_move = nullptr;
 7 }
 8 ChessGameLL::~ChessGameLL() {
 9     Node *curr, *nextone;
10     curr = head;
11     while (curr) {
12         nextone = curr->next;
13         delete curr;
14         curr = nextone;
15     }
16     head = current_move = nullptr;
17 }
18
19 bool ChessGameLL::empty() const { return !head; }
20 void ChessGameLL::push_back(Turn turn) {
```

```

21     Node *newnode = new Node;
22     newnode->data = turn;
23     Node *curr = head;
24     newnode->prev = nullptr;
25     newnode->next = nullptr;
26     if (empty()) {
27         head = newnode;
28         current_move = newnode;
29     } else {
30         while (curr->next) {
31             curr = curr->next;
32         }
33         curr->next = newnode;
34         newnode->prev = curr;
35         current_move = newnode;
36     }
37 }
38 void ChessGameLL::next_move() {
39     if(current_move->next == nullptr){
40         std::cout << "No next move...\n";
41         winner();
42     }
43     else{
44         if(current_move){
45             current_move = current_move->next;
46             std::cout << "Turn " << current_move->data.id << ": "
47             << ((current_move->data.color == 'W') ? "White" : "Black") << " - " << cu
48             << std::endl << std::endl;
49         }
50     }
51 }
52
53 void ChessGameLL::prev_move() {
54     if (current_move == head) {
55         std::cout << "No previous move....\n\n";
56         return;
57     } else {
58         current_move = current_move->prev;
59         std::cout << "Turn " << current_move->data.id << ": "
60         << ((current_move->data.color == 'W') ? "White" : "Black") << " - "
61         << current_move->data.move
62         << std::endl << std::endl;
63     }
64 }
65
66 void ChessGameLL::firstmove(){
67     current_move = head;
68     std::cout << "Turn " << current_move->data.id << ": "
69     << ((current_move->data.color == 'W') ? "White" : "Black") << " - " << curr
70     << std::endl << std::endl;
71 }
72
73
74 void ChessGameLL::winner() {
75     while (current_move->next) {
76         current_move = current_move->next;
77     }
78     std::cout << "Game ended with " << ((current_move->data.color == 'W') ? "White"
79     : "Black") << " Winning\n\n";
80 }

```

\\033[01;34m\\w\\033[00m]\$ g++ main.cpp ChessGameLL.cpp -o chess  
 \\033[01;34m\\w\\033[00m]\$ -o chess  
 sh: 7: -o: not found  
 \\033[01;34m\\w\\033[00m]\$ ./chess  
 Enter a game to review: foolsmate.dat  
  
 ---Game Start---

Turn 1: White - f3

Enter n/p/q: p  
No previous move....

Enter n/p/q: n  
Turn 1: Black - e5

Enter n/p/q: p  
Turn 1: White - f3

Enter n/p/q: n  
Turn 1: Black - e5

Enter n/p/q: n  
Turn 2: White - g4

Enter n/p/q: n  
Turn 2: Black - Qh4#

Enter n/p/q: n  
No next move...  
Game ended with Black Winning

Enter n/p/q: p  
Turn 2: White - g4

Enter n/p/q: q  
--- Ending Review ---  
\\033[01;34m\\w\\033[00m\\\$ ./chess  
Enter a game to review: foolsmate.dat

---Game Start---  
Turn 1: White - f3

Enter n/p/q: n  
Turn 1: Black - e5

Enter n/p/q: n  
Turn 2: White - g4

Enter n/p/q: n  
Turn 2: Black - Qh4#

Enter n/p/q: n  
No next move...  
Game ended with Black Winning

Enter n/p/q: q  
--- Ending Review ---  
\\033[01;34m\\w\\033[00m\\\$ ./chess  
Enter a game to review: game1.dat

---Game Start---  
Turn 1: White - e4

Enter n/p/q: p  
No previous move....

Enter n/p/q: n  
Turn 1: Black - c5

Enter n/p/q: n  
Turn 2: White - Nf3

Enter n/p/q: n  
Turn 2: Black - d6

Enter n/p/q: q  
--- Ending Review ---

\\033[01;34m\\w\\033[00m\\\$ exit

Script done on 2023-12-05 19:05:16+00:00 [COMMAND\_EXIT\_CODE="0"]