

Branch: master ▾

Find file

Copy path

[LeeByoungmo_CIS_17c_47698](#) / [Proj](#) / [texasHoldem_17C_Ver1.3](#) / [Hands.cpp](#) libmo No commit message

024972d 1 hour ago

[0 contributors](#)

Raw Blame History



186 lines (169 sloc) 5.27 KB

```
1  /*
2   * File:   Hands.cpp
3   * Author: Byoung Mo Lee
4   * Created on May 18, 2019 20:43 PM
5   * Purpose: Hands Class for Texas Holdem
6   *
7   */
8  #include <iostream>
9  #include <iomanip>
10 #include <set>
11 #include <list>
12 #include <algorithm>
13 #include <iterator>
14 #include <map>
15 #include <string>
16 #include "Hands.h"
17
18 using namespace std;
19
20 Hands::Hands():Player(){
21     this->Hands::setPlayerAct();
22     this->Hands::setInThePot();
23 }
24
25 string Hands::getHandsName(){
26     switch (this->Hands::getHands()) {
27         case 0: this->handsName = "Highcard";break;
28         case 1: this->handsName = "One Pair";break;
29         case 2: this->handsName = "Two Pair";break;
30         case 3: this->handsName = "Three of a card";break;
31         case 4: this->handsName = "Straight";break;
32         case 5: this->handsName = "Flush";break;
33         case 6: this->handsName = "Full house";break;
34         case 7: this->handsName = "Four of a card";break;
35         case 8: this->handsName = "Straight Flush";break;
36         default: this->handsName = "Bad Value";
37     }
38     return this->handsName;
39 }
40 int Hands::getHands(){
41     this->checkStraight();
42     this->Hands::setSuits();
43     this->Hands::setFaces();
44
45     if(this->isStraight!=0&&this->isFlush!=0){
46         this->isStFl=1;
47         this->hands=8;
48         set<int,greater<int>> stFl;
49         for(auto& it: this->Hands::getMyCards()){
50             if(it.getSuit()==this->isFlush) {stFl.insert(it.getFace());}
51         }
```

```

52     if(this->isStraight==15){
53         for(int i=14;i>9;i--){
54             if(stFl.find(i)==stFl.end()) this->isStFl=0;
55         }
56     }
57     else if(this->isStraight==15){
58         for(int i=5;i>1;i--){
59             if(stFl.find(i)==stFl.end()) this->isStFl=0;
60         }
61     }
62     else{
63         for(int i=this->isStraight;i>this->isStraight-5;i--){
64             if(stFl.find(i)==stFl.end()) this->isStFl=0;
65         }
66     }
67     if(this->isStFl==0){
68         this->hands=5;
69     }
70 }
71 else if(this->Hands::isPair==7) {this->hands=7;}
72 else if(this->isPair==6) {this->hands=6;}
73 else if(this->isFlush!=0) {this->hands=5;}
74 else if(this->isStraight!=0) {this->hands=4;}
75 else if(this->isPair==3) {this->hands=3;}
76 else if(this->isPair==2) {this->hands=2;}
77 else if(this->isPair==1) {this->hands=1;}
78 else {this->hands=0;}
79 // cout << endl;
80 // for(auto& it: this->Hands::getMyCards()){
81 //     it.toString();
82 // }
83 // cout << endl << "Hands=" << this->Hands::getHandsName() << endl;
84 return this->hands;
85 }
86 void Hands::checkStraight(){
87     this->isStraight=0;
88     set<int, greater<int>> faceSet;
89     int cnt=0;
90     for(auto& it: this->Hands::getMyCards()){
91         faceSet.insert(it.getFace());
92     }
93 // cout << endl;
94
95     set<int>::iterator it=faceSet.begin();
96
97     while(it != faceSet.end()&&cnt!=4){
98         if(*it-*faceSet.upper_bound(*it)==1) {cnt++;}
99         else{cnt=0;}
100 // cout << *it << ' ';
101 ++it;
102 }
103 if(cnt==4&&*it+4!=14) {this->isStraight=*it+4;}
104 else if(cnt==4&&*it+4==14) {this->isStraight=15;}
105 if(*it==14){
106     while(it != faceSet.end()&&cnt!=3){
107         if(*it-*faceSet.upper_bound(*it)==1) {cnt++;}
108         else{cnt=0;}
109 // cout << *it << ' ';
110 ++it;
111 }
112 }
113 if(cnt==3&&*it+3==5) {this->isStraight=14;}
114 }
115
116
117 void Hands::setSuits(){

```

```

118     this->isFlush=0;
119     multiset<int> suitSet;
120     for(auto& it: this->Hands::getMyCards()){
121         suitSet.insert(it.getSuit());
122     }
123     // for(int elem:suitSet){
124     //     cout << elem << ' ';
125     // }
126
127     for(int i=0;i<4;i++){
128         if(suitSet.count(i)>4) {
129             this->suits.insert(pair<int,int>(suitSet.count(i),i));
130             this->isFlush=1;
131         }
132     }
133     // cout << endl;
134     map<int,int>::iterator pos;
135     // for(pos=this->suits.begin();pos!=this->suits.end();++pos){
136     //     cout << "num: " << pos->first << ' '
137     //         << "suits: " << pos->second << endl;
138     // }
139     //
140     // cout << endl;
141
142
143 }
144 void Hands::setFaces(){
145     this->isPair=0;
146     multiset<int> faceSet;
147     for(auto& it: this->Hands::getMyCards()){
148         faceSet.insert(it.getFace());
149     }
150     // for(int elem:faceSet){
151     //     cout << elem << ' ';
152     // }
153     for(int i=2;i<15;i++){
154         if(faceSet.count(i)>1) this->faces.insert(pair<int,int>(faceSet.count(i),i));
155     }
156     // cout << endl;
157     map<int,int>::iterator pos;
158     // for(pos=this->faces.begin();pos!=this->faces.end();++pos){
159     //     cout << "pairs: " << pos->first << ' '
160     //         << "faces: " << pos->second << endl;
161     // }
162     // cout << endl;
163     // cout << endl << "faces.empty? " << faces.empty() << endl;
164     // cout << endl << "faces.size()? " << faces.size() << endl;
165
166     if(this->faces.empty()) {this->isPair=0;}
167     else if(this->faces.size()==1){
168         pos=this->faces.begin();
169         // cout << "pos->first: " << pos->first << endl;
170         if(pos->first==2) this->isPair=1;
171         else if(pos->first==3) this->isPair=3;
172         else if(pos->first==4) this->isPair=7;
173     }
174
175     else if(this->faces.size()>1){
176         pos=this->faces.begin();
177
178         // cout << "pos2->first: " << pos->first << endl;
179         if(pos->first==2) this->isPair=2;
180         else if(pos->first==3) this->isPair=6;
181     }
182
183

```

```
184 //    cout << "isPair: " << this->isPair << endl;  
185  
186 }
```